Betriebsanleitung

XCx-micro Steuereinheiten

XCS20C / XCS20P

Betriebsanleitung XCx-micro Steuereinheiten Version 03/08
Artikel-Nr. R4.322.2260.0 (322 385 40)

schleicher
control systems

Zweck der Betriebsanleitung

Die vorliegende Betriebsanleitung dient als Anleitung zur Projektierung, Programmierung, Installation, Inbetriebnahme, Betrieb und Wartung der XCS20C / XCS20P.

Die Betriebsanleitung enthält Projektierungs-, Programmier-, Bedienungshinweise und technische Daten der XCS20C / XCS20P.

Zielgruppe

Die Betriebsanleitung ist für geschulte Fachkräfte ausgelegt. Es werden besondere Anforderungen an die Auswahl und Ausbildung des Personals gestellt, die mit dem Automatisierungssystem umgehen. Als Personen kommen z.B. Elektrofachkräfte und Elektroingenieure in Frage, die entsprechend geschult sind (siehe auch Sicherheitshinweise "Personalauswahl und -qualifikation").

Gültigkeit der Betriebsanleitung

XCS20C ab Modulrevision A-02

XCS20P ab Modulrevision A-01

Software-Version V.06.23/3

Vorgängerversion der Betriebsanleitung

01/06 05/06 09/06 04/07 02/08

Bezugsmöglichkeiten für Betriebsanleitungen

Alle Betriebsanleitungen können kostenlos vom Internet: *http://www.schleicher-electronic.com* geladen, oder unter Angabe der Artikel-Nr. bestellt werden bei:

SCHLEICHER Electronic
GmbH & Co. KG
Pichelswerderstraße 3-5
D-13597 Berlin

Copyright by

SCHLEICHER Electronic
GmbH & Co. KG
Pichelswerderstraße 3-5
D-13597 Berlin

Telefon +49 30 33005-330
Telefax +49 30 33005-305
Hotline +49 30 33005-304
Internet *http://www.schleicher-electronic.com*

Änderungen und Irrtum vorbehalten

Inhaltsverzeichnis

1 Übersicht 6
2 Netzteil und Steuereinheiten 9
2.1 Netzteil XNT24 9
2.1.1 Anschlüsse XNT24 9
2.1.2 Digitale Eingänge XNT24 10
2.2 Steuereinheit XCS20C / XCS20P 11
2.2.1 Anschlüsse XCS20C 12
2.2.2 LED-Anzeigen XCS20C 12
2.2.3 Anschlüsse XCS20P 13
2.2.4 LED-Anzeigen XCS20P 14
2.2.5 Reset-Taster und Betriebsartenschalter XCS20C / XCS20P 15
2.2.6 Serielle Schnittstellen XCS20C / XCS20P 17
3 Installation 18
3.1 Mechanische Installation 18
3.1.1 Montagelage 18
3.1.2 Montageabstände 18
3.1.3 Tragschienenmontage 19
3.2 Elektrische Installation 20
4 Die SPS 21
4.1 Programmierung 21
4.1.1 Anschluss eines PC als Programmiergerät 21
4.1.2 Verbindungskabel 21
4.2 Speicheraufbau 22
4.2.1 Programmspeicher 22
4.2.2 Pufferspeicher für interne Systemdaten 22
4.2.3 Prozessdatenspeicher (Prozessabbildspeicher) 22
4.3 Speicherpufferung 22
4.4 Das Multi-Task-System 23
4.4.1 Anwender-Tasks 23
4.4.2 Systemtasks und SPGs beim Auftreten von Betriebssystemfehlern 26
4.4.3 Deklarieren von Anwender-Tasks 27
4.4.4 Prioritäten der Anwender-Tasks 27
4.4.5 Anwender-Task-Informationen 28
4.4.6 Tasks und Watchdogs 30
4.4.7 Tasks und Programme 31
4.5 SPS-Betriebszustände und Startverhalten 32
4.5.1 Betriebszustände 32
4.5.2 Wechseln der Betriebszuständen mit MULTIPROG 33
4.5.3 Startverhalten der SPS nach dem Einschalten der Versorgungsspannung 34
4.6 SPS-Adressen 35
4.6.1 Diagnosedaten und Prozessabbild 35
4.6.2 Systemmerker 36
4.6.3 Feldbusspezifische SPS-Adressen 37
4.7 Konfigurieren der SPS 38
4.7.1 Eingänge, Ausgänge und der Prozess 38
4.7.2 Editieren der I/O-Konfiguration in MULTIPROG 38
4.8 Setzen und Auslesen der Echtzeituhr 42
4.9 Interrupt-Betrieb 43
4.10 Parametrier- und Diagnosefunktionen über die SPS 44
4.10.1 Übersicht 44
4.10.2 Adressierung und Datenaufbau 45
4.10.3 Ablauf 45
4.10.4 Funktion 0 Sammelfehler auslesen 46
4.10.5 Funktion 1 Modul-Spannungsversorgung überwachen 47
4.10.6 Funktion 2 Überlast der Ausgangstreiber feststellen 48
4.10.7 Funktion 3 Prozessdatenbreite ermitteln 49
4.10.8 Funktion 4 Modul-Konfiguration auslesen 50
4.10.9 Funktion 8 Firmware-Version auslesen 51

4.10.10 Funktion 17 Busadresse auslesen ... 51
4.10.11 Funktion 112 und 113 Baudraten der seriellen Schnittstellen einstellen ... 52
4.10.12 Funktion 114 Boot-Up-Delay einstellen ... 52
4.10.13 Funktion 115 SPS-Reboot ... 53
4.10.14 Funktion 116 CANopen-Task deaktivieren ... 53
4.10.15 Funktion 117 Zykluszeit für synchron gekoppelte Tasks einstellen ... 54
4.10.16 Funktion 126 Baudrate für CANopen einstellen ... 54
4.10.17 Funktion 127 Busadresse einstellen ... 55
4.10.18 Funktion 255 Reset ... 55
4.11 Interaktives Diagnose - Interface ... 56
4.11.1 Übersicht ... 56
4.11.2 Einstellungen ... 56
4.11.3 Eingabe der PDF - Codes ... 56
4.12 Clear User - Flash ... 57
4.12.1 Beschreibung ... 57
4.13 Konformität gemäß IEC 61131-3 ... 58
5 Feldbus CANopen ... 60
5.1 Grundlagen ... 60
5.1.1 Belegung der Prozessdatenobjekte (PDO) ... 60
5.1.2 Nodeguarding ... 60
5.1.3 Lifeguarding ... 60
5.2 CANopen-Konfigurationsdaten ... 61
5.2.1 EDS-Dateien ... 61
5.3 Verkabelung CANopen ... 61
5.3.1 Abschlusswiderstände CANopen ... 61
5.4 Einstellen der Knotennummern CANopen ... 61
5.5 Einstellen der Datenübertragungsrate CANopen ... 62
6 Feldbus PROFIBUS-DP ... 63
6.1 Bustopologie PROFIBUS-DP ... 63
6.2 Aufbaurichtlinien für PROFIBUS-Netze ... 64
6.3 Buskabel PROFIBUS-DP ... 65
6.4 Bussegmentlänge PROFIBUS-DP ... 65
6.5 Busverkabelung PROFIBUS-DP ... 66
6.6 Projektierung PROFIBUS-DP ... 68
6.7 Inbetriebnahme PROFIBUS-DP ... 69
6.8 Diagnose am PROFIBUS-DP ... 69
6.9 Reaktionszeiten PROFIBUS-DP ... 71
7 Fehlermeldungen ... 72
8 Technische Daten und Abmessungen ... 76
9 Anhang ... 79
9.1 Warenzeichenvermerke ... 79
10 Sicherheitshinweise ... 80
10.1 Bestimmungsgemäße Verwendung ... 80
10.2 Personalauswahl und -qualifikation ... 80
10.3 Projektierung, Programmierung, Installation, Inbetriebnahme und Betrieb ... 81
10.4 Wartung und Instandhaltung ... 81
10.5 Gefahren durch elektrische Energie ... 81
10.6 Umgang mit verbrauchten Batterien ... 81
11 Abbildungsverzeichnis ... 82
12 Index ... 82

Darstellungskonventionen

Sicherheits- und Handhabungshinweise werden in dieser Programmieranleitung durch besondere Kennzeichnungen hervorgehoben:

Bedeutet, dass Personen, das Automatisierungssystem oder eine Sache beschädigt werden kann, wenn die entsprechenden Hinweise nicht eingehalten werden.

Hebt eine wichtige Information hervor, die die Handhabung des Automatisierungssystems oder den jeweiligen Teil der Betriebsanleitung betrifft.

Weitere Objekte werden folgendermaßen dargestellt.

Objekt	Beispiel
Dateinamen	HANDBUCH.DOC
Menüs / Menüpunkte	*Einfügen / Graphik / Aus Datei*
Pfade / Verzeichnisse	*C:\Windows\System*
Hyperlinks	*http://www.schleicher-electronic.com*
Programmlisten	`MaxTsdr_9.6  = 60` `MaxTsdr_93.75 = 60`
Tasten	<Esc> <Enter> (nacheinander drücken) <Ctrl+Alt+Del> (gleichzeitig drücken)
Bezeichner der Konfigurationsdaten	Q34 und Q.054
Namen der Koppelspeicher-Variablen	*`cncMem.sysSect[n].flgN2P.bM345Act`*

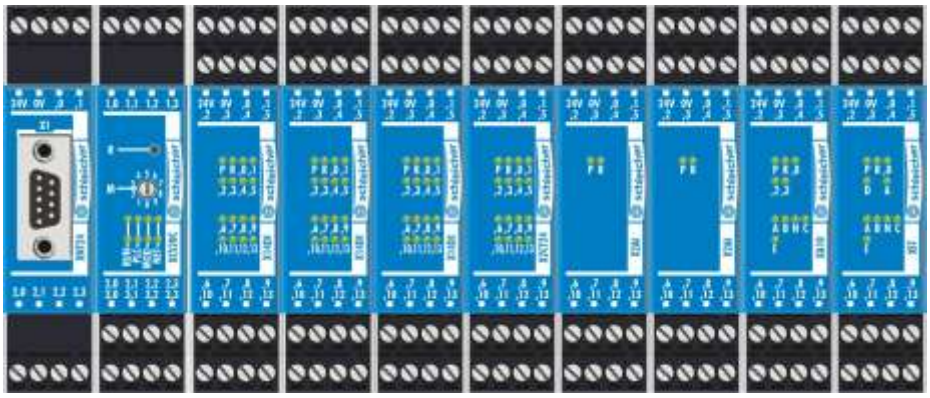
1 Übersicht

Mit den Steuereinheiten können Kleinsteuerungen im System XCx-micro aufgebaut werden.

Die Kleinsteuerungen bestehen immer aus:

- einem Netzteil
- einer Steuereinheit
- bis zu acht Ein-/Ausgangsmodulen

Sie werden auf einer 35 mm Tragschiene befestigt und über einen seitlich am Modul angebrachten Steckverbinder miteinander verbunden.



Das Betriebssystem unterstützt bis zu 16 Ein- und Ausgangsmodule.

Netzteil DC 24 V	
	XNT24 mit einer seriellen Schnittstelle und zwei digitalen Eingängen

Steuereinheiten

mit Feldbusanschluss CANopen und serieller Schnittstelle

XCS20C

- Programmierung nach IEC 61131-3
- 1 MB Speicher
- Motorola MC68LC302 CPU
- Prozessortakt 24 MHz
- Multitask-Betriebssystem
- Batteriegepufferte Echtzeit-Uhr
- zwei serielle Schnittstellen RS232 (die erste serielle Schnittstelle ist auf dem Netzteil herausgeführt)
- Feldbusanschluss CANopen

mit Feldbusanschluss PROFIBUS und serieller Schnittstelle

XCS20P

- Programmierung nach IEC 61131-3
- 1 MB Speicher
- Motorola MC68LC302 CPU
- Prozessortakt 24 MHz
- Multitask-Betriebssystem
- Batteriegepufferte Echtzeit-Uhr
- zwei serielle Schnittstellen RS232 (die erste serielle Schnittstelle ist auf dem Netzteil herausgeführt)
- Feldbusanschluss PROFIBUS

Verwendbare I/O-Module aus dem System XCx-micro

Die Module werden in dieser Anleitung nicht beschrieben. Siehe dazu Betriebsanleitung XCx-micro I/O-Module Artikel-Nr. R4.322.2280.0 (322 385 44).

Digitalmodule

X14DI

- 14 digitale Eingänge DC 24 V
- entsprechend IEC61131-2 Typ I

X14DO

- 14 digitale Ausgänge DC 24 V / 0,5 A
- entsprechend IEC61131-2

Digitalmodule

X6DI8DIO

6 digitale Eingänge
8 digitale Kombi-I/O
Alle Kombi I/O sind als Eingänge DC 24 V oder Ausgänge 0,5 A nutzbar.

Analogmodule

X2AI

2 analoge Eingangs-Kanäle
±10 V oder 0..20 mA
Auflösung 12 Bit

X2AO

2 analoge Ausgangs-Kanäle
±10 V oder 0..20 mA
Auflösung 12 Bit

Zählermodule

X2CT24

2 Vor-/Rückwärtszähler, 32 Bit
mit Zähleingängen 24 V

X2CT05

2 Vor-/Rückwärtszähler, 32 Bit
mit Zähleingängen 5 V

2 Netzteil und Steuereinheiten

2.1 Netzteil XNT24

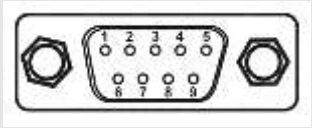
Das Netzteil wird mit DC 24 V betrieben, es stellt die benötigten Spannungen für die Steuereinheit zur Verfügung.

Es ist zusätzlich ausgerüstet mit der ersten seriellen Schnittstelle der Steuereinheit und zwei digitalen Eingängen.

Abbildung 1: Frontansicht XNT24

2.1.1 Anschlüsse XNT24

Klemme			
24V	Versorgungsspannung DC 24 V		
0V	Versorgungsspannung 0 V		
.0	digitaler Eingang 0		siehe unten
.1	digitaler Eingang 1		
2.0	GND	Ground	erste serielle Schnittstelle RS232 der Steuereinheit Die serielle Schnittstelle ist parallel auf dem 9-poligen D-Sub Steckverbinder herausgeführt.
2.1	TxD	Transmit Data	
2.2	RxD	Receive Data	
2.3	SHLD	Schirm	

D-Sub, 9-polig, Stecker				
	Pin	Signal		Erläuterung
 Abbildung 2: Serielle Schnittstelle XNT24	1	nc	not connected	nicht angeschlossen
	2	RxD	Receive Data	Eingangsdaten
	3	TxD	Transmit Data	Ausgangsdaten
	4	DTR	Data Terminal Ready	interne Brücke nach Pin 6
	5	GND	Logic Ground	nicht für Schirm
	6	DSR	Data Set Ready	interne Brücke nach Pin 4
	7	RTS	Request To Send	interne Brücke nach Pin 8
	8	CTS	Clear To Send	interne Brücke nach Pin 7
	9	nc	not connected	nicht angeschlossen
	Die serielle Schnittstelle ist parallel auf den Klemmen 2.0 bis 2.3 herausgeführt.			

2.1.2 Digitale Eingänge XNT24

Die beiden schnellen digitalen Eingänge können im SPS-Programm verarbeitet werden. Wegen ihrer kurzen Reaktionszeiten, unabhängig von der Belastung der Steuereinheit, sind sie für zeitkritische Anwendungen einsetzbar. Siehe hierzu auch Kap. "Interrupt-Betrieb", Seite 43.

Kennwerte siehe Technische Daten Seite 76.

In den mit MULTIPROG gelieferten Templates für die XCS20-Steuereinheiten sind die Variablen bereits angelegt.
Arbeitsblatt Global_Variables, Gruppe XFIO_Variables

Variable	Adresse	Klemme	Eingang Nr.
xfio_in0	% IX 10000.0	.0	0
xfio_in1	% IX 10000.1	.1	1

2.2 Steuereinheit XCS20C / XCS20P

Die Steuereinheit XCS20C ist eine SPS mit CANopen Funktionalität.
Die Steuereinheit XCS20P ist eine SPS mit PROFIBUS-DP Slave Funktionalität.

Abbildung 3: Frontansicht XCS20C

Abbildung 4: Frontansicht XCS20P

Die Steuereinheit besitzt folgende Merkmale:

- Programmierung nach IEC 61131-3
- 1 MB Speicher
- Motorola MC68LC302 CPU
- Prozessortakt 24 MHz
- Multitask-Betriebssystem
- Batteriegepufferte Echtzeit-Uhr
- zwei serielle Schnittstellen RS232 (die erste serielle Schnittstelle ist auf dem Netzteil herausgeführt)
- Feldbusanschluss CANopen bzw. PROFIBUS

2.2.1 Anschlüsse XCS20C

Klemme			
1.0	SHLD	Schirm	zweite serielle Schnittstelle RS232
1.1	RxD	Receive Data	
1.2	TxD	Transmit Data	
1.3	GND	Ground	
2.0	DRAIN		Schnittstelle CANopen
2.1	CAN-L		
2.2	CAN-H		
2.3	GND	Ground	
3.0	DRAIN		Schnittstelle CANopen
3.1	CAN-L		
3.2	CAN-H		
3.3	GND	Ground	

Die Klemmen 2.x und 3.x sind je einzeln intern gebrückt. Also CAN-H mit CAN-H u.s.w.

2.2.2 LED-Anzeigen XCS20C

Die Anzeigen bestehen aus jeweils einer roten und einer grünen LED.

Abbildung 5: LEDs XCS20C

LED-Anzeigen				
	Bezeichnung	Farbe	Zustand	Bedeutung
	RUN	grün	an	Prozessor der Steuereinheit läuft
		rot	an	Betriebssystemfehler
	PLC	grün	an	SPS läuft, I/O-Refresh läuft
			blinkend	SPS-STOP, I/O-Refresh angehalten
			aus	nach SPS-Reset (kein aktives SPS-Programm)
		rot	an	--
			blinkend	SPS-Fehler
	MOD (Modulstatus)	grün	an	CAN-Stack ist initialisiert, Steuereinheit ist bereit
			blinkend	--
		rot	an	schwerer Fehler, kann vom Anwender nicht behoben werden
			blinkend	Versorgungsspannung DC 24 V fehlt oder Kurzschluss an den Ausgängen
	NET (Netzstatus)	grün	an	CAN State Operational
			blinkend	CAN State Pre-Operational
		rot	an	BUS OFF z.B. bei fehlerhafter Baudrate, Problemen der Busphysik (nicht aber vollst. Kabelbruch!), Hardwarefehler eines CAN-Controllers im Netzwerk
			blinkend	CAN-Fehler aufgetreten z.B. 24 V an CAN-Controller aber kein aktiver CAN-Controller am Bus

2.2.3 Anschlüsse XCS20P

Klemme			
1.0	SHLD	Schirm	
1.1	RxD	Receive Data	zweite serielle Schnittstelle RS232
1.2	TxD	Transmit Data	
1.3	GND	Ground	
2.0	SHLD	Schirm	
2.1	A (RxD/TxD-N)		PROFIBUS-Schnittstelle
2.2	B (RxD/TxD-P)		
2.3	GND	Ground	
3.0	TN	Termination N	
3.1	A (RxD/TxD-N)		PROFIBUS-Schnittstelle
3.2	B (RxD/TxD-P)		
3.3	TP	Termination P	

Die Abschlusswiderstände für die PROFIBUS-Schnittstelle sind im Modul vorhanden. Durch Brücken von 3.0 auf 3.1 sowie von 3.2 auf 3.3 können diese zugeschaltet werden.

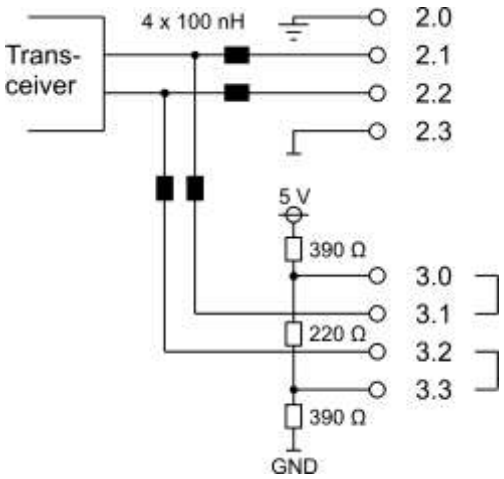


Abbildung 6: Anschlüsse XCS20P

2.2.4 LED-Anzeigen XCS20P

Die Anzeigen bestehen aus jeweils einer roten und einer grünen LED.

LED-Anzeigen

Abbildung 7: LEDs XCS20P

Bezeichnung	Farbe	Zustand	Bedeutung
RUN	grün	an	Prozessor der Steuereinheit läuft
	rot	an	Betriebssystemfehler
PLC	grün	an	SPS läuft, I/O-Refresh läuft
		blinkend	SPS-STOP, I/O-Refresh angehalten
		aus	nach SPS-Reset (kein aktives SPS-Programm)
	rot	an	--
		blinkend	SPS-Fehler
BF	grün	an	Normalbetrieb
	rot	an	Keine Busverbindung (bus fail) Bruch des Feldbuskabels oder der Master betreibt den Bus nicht (mehr).
DIA	grün	an	Normalbetrieb
	rot	an	Diagnosemeldung Die Steuereinheit hat eine PROFIBUS-DP Diagnosemeldung an den Master abgesetzt. (z.B. SPS-Stop)

2.2.5 Reset-Taster und Betriebsartenschalter XCS20C / XCS20P

Reset-Taster

Abbildung 8: RESET-Taster XCS20C / XCS20P

Der RESET-Taster löst einen Hardware-Reset aus, gleichbedeutend mit Power-Off.

Betriebsartenschalter

Abbildung 9: Drehschalter XCS20C / XCS20P

Stellung	Bezeichnung	
0		Test / Init, erweiterte Diagnose, SPS-Stop
1	PROG	Betriebsart Programmierung, SPS-Stop
2	WARM	Warmstart der SPS
3	COLD	Kaltstart der SPS (Reinitialisierung der Retainvariablen)
4		Update-Modus (für Sevicezwecke, Betriebssoftware-Update)
5	Ausgeben	Boot-Projekt und CAN Konfiguration
6	Einlesen	Boot-Projekt und CAN Konfiguration
7	WARM	Warmstart der SPS
8	WARM	Warmstart der SPS
9	WARM	Warmstart der SPS

Die Stellung des Schalters kann vom SPS-Programm aus abgefragt werden.

In den mit MULTIPROG gelieferten Templates für die XCS20-Steuereinheiten ist die Variable bereits angelegt.
Arbeitsblatt: Global_Variables, Gruppe: XFIO_Variables
Variable: **xfio_sb0**

Der Wert der Variablen entspricht der Schalterstellung 0 bis 9.

2.2.6 Serielle Schnittstellen XCS20C / XCS20P

Es stehen zwei serielle RS232 Schnittstellen, eine auf dem Netzteil, die zweite auf der Steuereinheit zur Verfügung.

Nach dem Einschalten der Steuereinheit sind die Schnittstellen für den Anschluss eines PC zur Programmierung (siehe Seite 21) mit MULTIPROG oder dem Anschluss eines Bediengerätes COP initialisiert.

Folgende Parameter sind eingestellt:

- Baudrate: 38400
- Datenbits: 8
- Stopbits: 1
- Parität: NO

Die Baudrate jeder Schnittstelle kann von der SPS mittels der Parametrier- und Diagnosefunktionen 112 und 113 eingestellt werden (siehe Seite 44).

Die seriellen Schnittstellen können auch vom SPS-Programm mittels des Funktionsbausteines* Serial benutzt werden.

Wird eine serielle Schnittstelle vom SPS-Programm verwendet, wird sie mit den bei PORT_OPEN übergebenen Parametern geöffnet. Nach dem Schließen mit PORT_CLOSE wird die serielle Schnittstelle reinitialisiert. Sie kann dann, mit der zuletzt eingestellten Baudrate, wie nach dem Einschalten benutzt werden.

*Funktionsbausteine im Programmiersystem MULTIPROG im Projektbaum unter Bibliotheken.

3 Installation

3.1 Mechanische Installation

3.1.1 Montagelage

Die senkrechte Einbaulage muss eingehalten werden, damit eine freie Luftzirkulation gewährleistet wird.

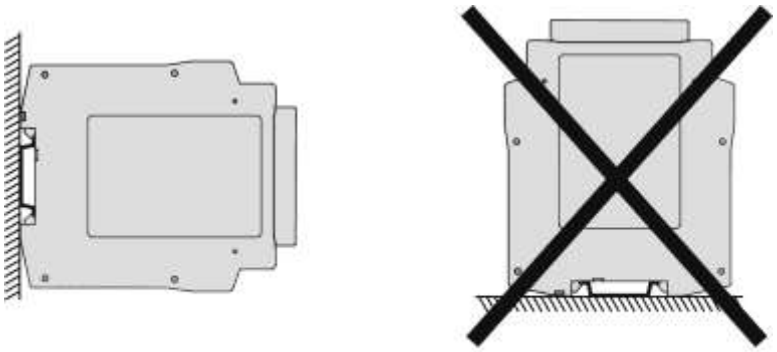


Abbildung 10: Montagelage

3.1.2 Montageabstände

Die Montageabstände von min. 50 mm müssen eingehalten werden, damit eine freie Luftzirkulation gewährleistet wird.

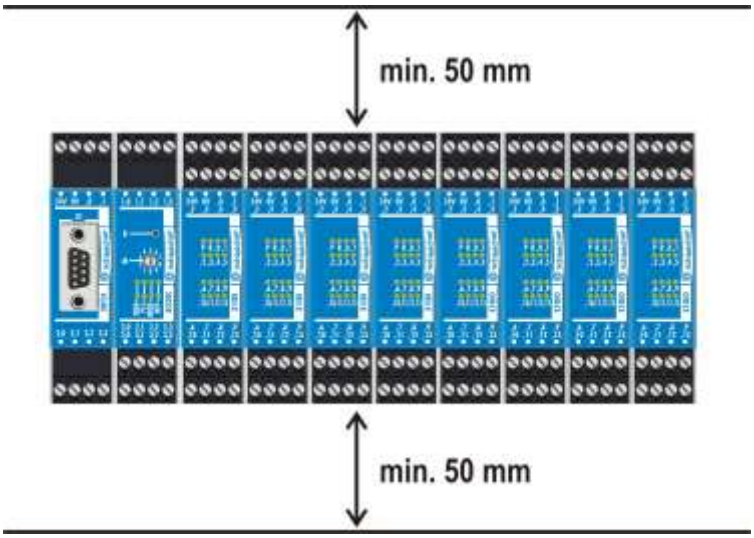
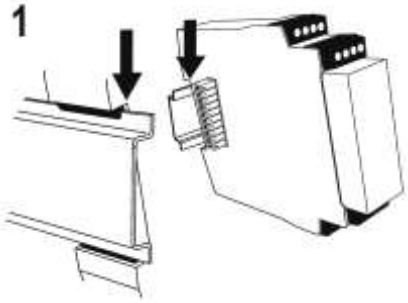
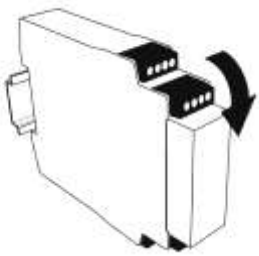
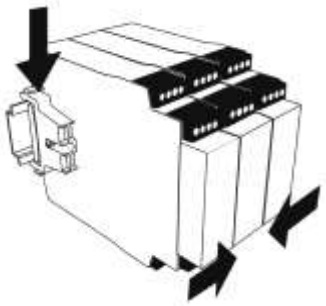
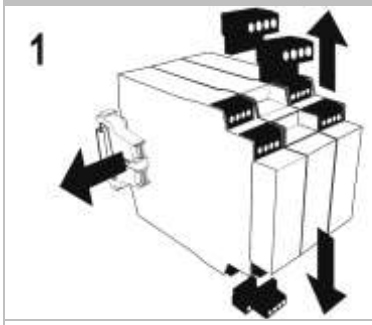
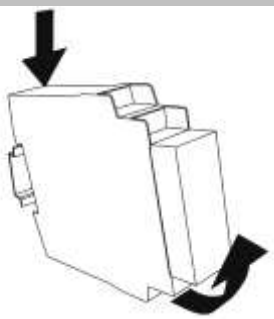


Abbildung 11: Montageabstände

Der Maximalausbau, Netzteil/Steuereinheit + 8 Module, soll eingehalten werden. Ein weiterer Ausbau wird aus EMV-Gründen nicht empfohlen.

3.1.3 Tragschienenmontage

Tragschiene Type TS 35mm/7,5 nach DIN EN 50022 verwenden.

Montage		
1 	2 	3 
Modul auf die Tragschiene einhängen. **Achtung** Dabei auf korrekten Sitz der Erdungsfeder achten. Die Erdungsfeder des Modules muss sicher und elektrisch gut leitend auf der Tragschiene aufliegen.	Durch leichten Druck in Pfeilrichtung auf die Tragschiene aufschnappen.	Wenn mehrere Module vorhanden sind - in Pfeilrichtung einzeln zusammenschieben, bis der seitliche Steckverbinder einrastet und Endklammern rechts und links installieren.
Demontage		
1 	2 	3 
Steckblockklemmen mit Verdrahtung und Endklammer entfernen.	**Achtung** Wenn mehrere Module vorhanden sind - in Pfeilrichtung einzeln auseinanderschieben, bis der seitliche Steckverbinder getrennt ist.	Modul hinten herunterdrücken. Im heruntergedrückten Zustand in Pfeilrichtung von der Tragschiene nehmen.

3.2 Elektrische Installation

Installationsrichtlinien

- Die Steuereinheiten sind in elektrischen Betriebsmittelräumen oder in geerdeten geschlossenen Gehäusen aus Metall (z.B. Schaltkasten, Schaltschrank) zu installieren. Die zur Aufnahme vorgesehene Tragschiene muss großflächig und gut leitend mit Masse verbunden werden.

Zum Schutz der Module vor Entladung statischer Elektrizität muss sich das Bedienpersonal vor dem Öffnen von Schaltkästen oder Schaltschränken elektrostatisch entladen.

- Alle digitalen und analogen I/O-Leitungen sind getrennt von DC/AC-Leitungen > 60 V zu verlegen.
- Schirmung:
  Störempfindliche Signalleitungen sind geschirmt auszuführen. Hierzu zählen Analogkabel, Datenverbindungskabel (z.B. RS232-Kabel) und Kabel für schnelle Digitalsignale (z.B. Kabel zu den Zählereingängen des X2CT24-Moduls).
  Die geschirmten Kabel sind über eine Schirmschiene in der unmittelbaren Nähe der Module zu führen. Dort wird ein Stück Isolierung am Kabel entfernt um das Schirmgeflecht (z.B. mit einer Kabelschelle aus Metall) großflächig auf der Schiene zu kontaktieren. Das Schirmgeflecht sollte hier nicht enden, sondern bis in die Nähe der Modulklemmen geführt werden. Statt auf einer Schirmschiene, die eine gute Hf-Verbindung zur Montageplatte haben muss, kann der Schirm auch direkt auf die metallene Montageplatte der Module aufgelegt werden (PE-Potential).
  Die wirkungsvollste Schirmung ergibt sich, wenn die Kabel beidseitig geerdet werden.

Bei Modulen mit digitalen Kombikanälen ist darauf zu achten, dass das Anlegen von 24 V an den Kombikanal, ohne Einspeisung der Versorgungsspannung unzulässig ist.
Sonst kommt es über die Ausgangsschaltung des Moduls zur Rückspeisung der Versorgungsspannung, in deren Folge eine Fehlfunktion oder Zerstörung der Ausgangsschaltung auftreten kann.

Daher bei Not-Aus Eingangssignale und Versorgungsspannung gleichzeitig abschalten.

Sinngemäß betreffen die oben gemachten Aussagen auch digitale Ausgangskanäle, wenn sie in fehlerhafter Weise mit 24 V beschaltet werden.

4 Die SPS

4.1 Programmierung

Die Programmierung der Steuereinheit erfolgt mit der Programmiersoftware MULTIPROG auf einem PC.

4.1.1 Anschluss eines PC als Programmiergerät

Der PC muss mit einer freien seriellen Schnittstelle RS232 ausgerüstet sein.
Baudrate: 9600, 19200, 38400 (default), 57600
Datenbits: 8
Stopbits: 1
Parität: NO

Der Anschluss kann wahlweise an der ersten (auf dem Netzteil) oder der zweiten seriellen Schnittstelle der Steuereinheit erfolgen.

4.1.2 Verbindungskabel

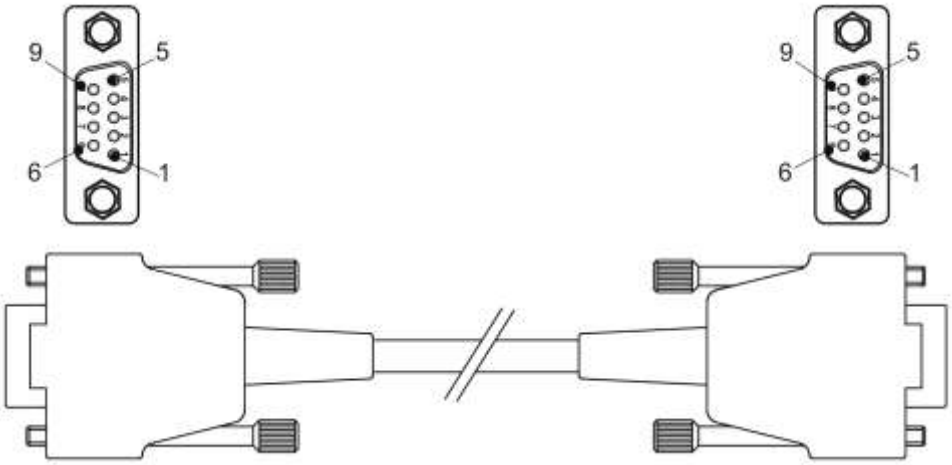


Abbildung 12: Verbindungskabel Steuereinheit-PC

Verbindungskabel erste serielle Schnittstelle - PC			
Signal	D-Sub, 9-pol. Buchse	D-Sub, 9-pol. Buchse	Signal
nicht angeschlossen	1	1	nicht angeschlossen
RxD	2	3	TxD
TxD	3	2	RxD
DTR	4	6	DSR
GND	5	5	GND
DSR	6	4	DTR
RTS	7	7	RTS
CTS	8	8	CTS
nicht angeschlossen	9	9	nicht angeschlossen

4.2 Speicheraufbau

Die Steuereinheit verfügt über mehrere Typen physikalischer Speicher für unterschiedliche Zwecke.

Um ein SPS-Programm so zu speichern, dass es nach Abschalten der SPS oder nach einem Stromausfall noch verfügbar ist, kann es auf dem Flash-Speicher geschrieben werden.

Um remanente Variablen zu speichern, wird das batteriegepufferte RAM verwendet, wobei dieser Speichertyp kurze Zugriffszeiten und nicht flüchtiges Speichern ermöglicht.

Der Laufzeitspeicher ist der Speicher, in dem das gesamte SPS-Programm und alle dazugehörigen Daten während der Laufzeit gespeichert werden.
Der Laufzeitspeicher setzt sich aus mehreren Bestandteilen zusammen, die für verschiedene Typen von Daten vorgesehen sind.

SPS-Programm	Programmspeicher

Buffer	Pufferspeicher für interne Systemdaten
I	Prozessdatenspeicher
Q	
M	

4.2.1 Programmspeicher

Im Speicher für das SPS-Programm werden alle Teile des SPS-Programms gespeichert, die während der Laufzeit verwendet werden. Das bedeutet alle Programme, Funktionsbausteine und Funktionen, Tasks und Konfigurationsdaten.

4.2.2 Pufferspeicher für interne Systemdaten

Der Puffer ist ein interner Teil des Speichers, in dem die SPS alle internen Systemdaten speichert. Die SPS verwendet diesen Puffer z.B. für die Kommunikation mit MULTIPROG. Beispielsweise werden Fehlermeldungen, die an MULTIPROG übermittelt werden, im Puffer gespeichert.

4.2.3 Prozessdatenspeicher (Prozessabbildspeicher)

Im Prozessdatenspeicher werden alle Daten gespeichert. Der Prozessdatenspeicher kann in verschiedene Bereiche eingeteilt werden.

- Speicherbereich 'I' für Eingänge, die vom Prozess gelesen werden.
- Speicherbereich 'Q' für Ausgänge, die in den Prozess geschrieben werden.
- Speicherbereich 'M' für alle Variablen, die weder Eingänge noch Ausgänge sind.

Der Speicher für Eingänge und Ausgänge heißt auch Prozessabbildspeicher.

4.3 Speicherpufferung

Der Akku der Steuereinheit puffert den RAM-Speicher sowie die Echtzeit-Uhr. Es wird eine wiederaufladbare Vanadium-Pentoxid-Lithium-Zelle mit einer Pufferzeit von mindestens 3 Monaten verwendet. Um die Pufferzeit von 3 Monaten zu erreichen, muss die Zelle voll aufgeladen sein. Das wird mit einer ununterbrochenen Betriebszeit der Steuereinheit von 4 h erreicht wird.

4.4 Das Multi-Task-System

Basis des SPS-Laufzeitsystemes ist ein Echtzeit-Betriebssystem, das durch Taskprioritäten gesteuert wird. Drei Prioritätsstufen für Tasks werden verwendet:

- Prioritätsstufe für Überwachungstasks (supervisor tasks)
- Prioritätsstufe für Anwender-Tasks (user tasks)
- Prioritätsstufe für Betriebssystem-Tasks (system tasks)

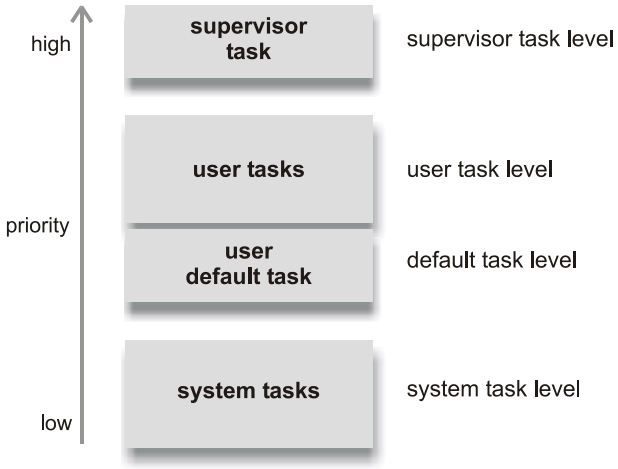


Abbildung 13: Multi-Task-System

Auf der Prioritätsstufe für Betriebssystem-Tasks laufen Tasks vom Anwender unbeeinflusst ab: Kommunikationstask, Debugtask, Speicherverwaltungstask und Systemkontrolltask.

Im Betriebssystem gibt es eine besonders geschützte Prioritätsstufe für die Überwachungstask. Die Überwachungstask ist eine Systemtask, die in der höchsten Prioritätsstufe abgearbeitet wird. Die Überwachungstask ermittelt Fehler, wie z.B. eine Division durch Null oder die Überschreitung der Ausführungszeit einer Task und aktiviert die entsprechende Systemtask.

4.4.1 Anwender-Tasks

Anwender-Tasks sind alle Tasks, die durch den Anwendungsprogrammierer festgelegt werden und sich auf die IEC 61131-3 Programmiersprachen stützen. Die Ausführung der Programme wird durch die Tasks gesteuert. Anwender-Tasks werden in der spezifischen Prioritätsstufe für Anwender-Tasks abgearbeitet. Das bedeutet, dass alle anwenderdefinierten Prioritäten nicht absolut, sondern relativ zur Prioritätsstufe für Anwender-Tasks sind. Die Tasks müssen in MULTIPROG eingerichtet werden. Die DEFAULT-Task gehört ebenfalls zur Prioritätsstufe für Anwender-Tasks. Sie ist die Anwender-Task mit der niedrigsten Priorität. Die DEFAULT-Task wird abgearbeitet, wenn zum entsprechenden Zeitpunkt keine Anwender-Task aktiv ist.

Anwender-Tasks müssen vom Programmierer deklariert werden. Es können verschiedene Anwender-Tasktypen verwendet werden.

DEFAULT-Task

Die DEFAULT-Task ist die Task mit der niedrigsten Priorität und wird abgearbeitet, wenn keine weitere Task aktiv ist. Die Default-Task ist eine zyklische Task. Das bedeutet, dass nach Beenden eines Zyklus der DEFAULT-Task immer der nächste Zyklus gestartet wird. Der Name der DEFAULT-Task ist DEFAULT. Um eine DEFAULT-Task in MULTIPROG einzufügen, geben Sie den Namen in Großbuchstaben ein und wählen als Tasktyp 'CYCLIC' aus.

Zyklische Tasks (CYCLIC)

Zyklische Tasks werden innerhalb eines festgelegten Zeitintervalls abgearbeitet. Sie werden gemäß ihrer Priorität abgearbeitet, die im Dialog 'Task ... einrichten' festgelegt werden muss.
Diesen Dialog können Sie über den Menüpunkt 'Einrichten...' aus dem Kontextmenü zur Task im Unterbaum 'Hardwarestruktur' des Projektbaums aufrufen. Der Bereich der Prioritätswerte liegt zwischen 0 und 16, wobei 0 die höchste Priorität ist.

Wenn die Watchdog-Zeit einer zyklischen Task höher ist als die eingestellte Intervallzeit und die Ausführung der Task nicht beendet ist, bevor die eingestellte Intervallzeit erreicht wird, werden ein oder mehrere Ausführungs-Zyklen übersprungen.

Ereignistasks (EVENT)

Ereignis-Tasks oder auch Event-Tasks werden vom Betriebssystem gestartet, wenn bestimmte Ereignisse auftreten.

Gegenwärtig sind folgende Ereignisse definiert.

Interne Bezeichnung	Ereignis-Nummer	Bemerkungen
PLC_EVENT_CAN	5	CANopen- Task
PLC_EVENT_MCSIO	8	MCS / XCS20 IO-Treiber Synchronisation (microLine, XCx micro)

Die Ereignisnummer wird in der Taskeinstellung von MULTIPROG verwendet, um das Ereignis zu spezifizieren, das die Ereignis-Task startet.

Die vorgegebene Priorität wird, außer bei gesetzter Bypass- Option vom System, berücksichtigt. (Bypass hebt den normalen Taskwechsel auf, so dass die zugewiesenen Programme sofort ausgeführt werden, wenn das Ereignis eintritt.)

Es werden bis zu 16 Ereignisse in eine Warteschlange gesetzt. Diese Ereignisse gehen daher nicht verloren und werden später ausgeführt. Dies gilt auch im Falle eines Auftretens neuer Ereignisse vor der Ausführung der zugewiesenen Ereignis-Task.

Systemtasks (SYSTEM)

System-Tasks bzw. Systemprogramme (SPG's) werden automatisch vom Betriebssystem gestartet, wenn im Zusammenhang mit dem Betriebssystem ein Ereignis auftritt.

System-Tasks werden nicht vom Watchdog überwacht.

Verschiedene SPG's sind verfügbar, wie in der folgenden Tabelle dargestellt:

Nr.	Name	Bedeutung	Aktionen
SPG 0	WARM_START	wird bei einem Warmstart ausgeführt	remanente Daten werden nicht initialisiert nicht-gepufferte Daten werden initialisiert die Open-Funktion des I/O-Treibers wird ausgeführt Anwender-Tasks werden aktiviert SPS wechselt in den Zustand 'Betrieb'
SPG 1	COLD_START	wird bei einem Kaltstart ausgeführt	alle Daten werden initialisiert die Open-Funktion des I/O-Treibers wird ausgeführt Anwender-Tasks werden aktiviert SPS wechselt in den Zustand 'Betrieb'
SPG 2	TO_STOP	wird ausgeführt, wenn die Programmausführung gestoppt wird	Anwender-Tasks werden deaktiviert alle Ausgänge werden aktualisiert die Close-Funktion des I/O-Treibers wird ausgeführt SPS wechselt in den Zustand 'STOP'
SPG 10	WATCHDOG	wird ausgeführt, wenn die Ausführung einer Task nicht innerhalb ihrer Watchdogzeit beendet ist	Anwender-Tasks werden deaktiviert alle Ausgänge werden aktualisiert die Close-Funktion des I/O-Treibers wird ausgeführt SPS wechselt in den Zustand 'STOP'
SPG 11	ZERODIV	wird ausgeführt, wenn während der Programmausführung eine Division durch Null aufgetreten ist	Anwender-Tasks werden deaktiviert alle Ausgänge werden aktualisiert die Close-Funktion des I/O-Treibers wird ausgeführt SPS wechselt in den Zustand 'STOP'
SPG 12	STACKOVER	wird ausgeführt, wenn ein Stacküberlauf aufgetreten ist. Wird nur ausgeführt, wenn das Kontrollkästchen 'Stack-Prüfung' im Dialog 'Ressource ... einrichten' in MULTIPROG aktiviert wurde.	Anwender-Tasks werden deaktiviert alle Ausgänge werden aktualisiert die Close-Funktion des I/O-Treibers wird ausgeführt SPS wechselt in den Zustand 'STOP'
SPG 13	BADCAL	wird ausgeführt, wenn eine herstellerspezifische POE aufgerufen wird, die nicht existiert	Anwender-Tasks werden deaktiviert alle Ausgänge werden aktualisiert die Close-Funktion des I/O-Treibers wird ausgeführt SPS wechselt in den Zustand 'STOP'
SPG 14	IOERROR	wird ausgeführt, wenn ein Fehler im I/O-Treiber auftritt, während der Prozess abläuft	SPS setzt Abarbeitung fort
SPG 16	MATHERR	wird ausgeführt, wenn ein Gleitkommafehler in einer arithmetischen Funktion auftritt	Anwender-Tasks werden deaktiviert alle Ausgänge werden aktualisiert die Close-Funktion des I/O-Treibers wird ausgeführt SPS wechselt in den Zustand 'STOP'
SPG 17	CPU_OVERLOAD	wird ausgeführt, wenn eine CPU-Überlastung auftritt	Anwender-Tasks werden deaktiviert alle Ausgänge werden aktualisiert die Close-Funktion des I/O-Treibers wird ausgeführt

Nr.	Name	Bedeutung	Aktionen
			SPS wechselt in den Zustand 'STOP'
SPG 18	INITIODRV_ERR	wird ausgeführt, wenn beim Initialisieren des I/O-Treibers während eines Kalt- oder Warmstarts ein Fehler auftritt	SPS wird nicht gestartet
SPG 19	BOUNDS_ERR	wird ausgeführt, wenn die Grenzen eines Felds oder einer Struktur überschritten wurden. Wird nur ausgeführt, wenn das Kontrollkästchen 'Index-Prüfung' oder das Kontrollkästchen 'Feldbegrenzungs-Prüfung' im Dialog 'Ressource ... einrichten' in MULTIPROG aktiviert wurde.	Anwender-Tasks werden deaktiviert alle Ausgänge werden aktualisiert die Close-Funktion des I/O-Treibers wird ausgeführt SPS wechselt in den Zustand 'STOP'
SPG 20	BUS_ERR	wird ausgeführt, wenn Variablen mit einem Datentyp ≥ 2 Bytes und ungeraden Adressen verwendet wurden oder wenn in MULTIPROG ein interner Fehler aufgetreten ist. Nur bei Motorola-Plattformen.	Anwender-Tasks werden deaktiviert alle Ausgänge werden aktualisiert die Close-Funktion des I/O-Treibers wird ausgeführt SPS wechselt in den Zustand 'STOP'
SPG 21	STRING_ERR	wird ausgeführt, wenn ein Fehler bei einer Zeichenfolge-Operation auftritt, z.B. wenn eine Zeichenfolge durch eine andere ersetzt werden sollte, aber nicht gefunden wurde.	Das Verhalten einer Zeichenfolge-Ausnahme hat sich geändert! In der Standardeinstellung wird nach dem Auftreten einer Zeichenfolge-Ausnahme das SPG 21 aufgerufen. Zusätzlich wird ein Eintrag in den Fehlerkatalog vorgenommen, der die Modul- und Zeilennummer enthält. Die SPS bleibt im 'RUN'-Status.

4.4.2 Systemtasks und SPGs beim Auftreten von Betriebssystemfehlern

Wenn ein Laufzeitfehler auftritt, wie z.B. eine Division durch Null oder ein Stacküberlauf wird das entsprechende SPG automatisch ausgeführt, ohne dass der Anwendungsprogrammierer etwas tun muss.

Der Funktionsbaustein CLR_OUT kann dazu verwendet werden, alle Ausgänge auf Null zu setzen. Die Funktionen COLD_RESTART, WARM_RESTART, HOT_RESTART und CONTINUE können zum Wiederanlauf oder zum Fortsetzen der Programmausführung verwendet werden. Diese Funktionen und Funktionsbausteine werden in der kontextsensitiven Hilfe beschrieben.

Um zusätzliche Funktionalitäten in zu editieren, führen Sie folgende grundlegende Schritte durch:

- Editieren eines Programms mit einem Programmcode, der die zusätzliche Funktionalität im Unterbaum 'Logische POEs' des Projektbaums enthält.
- Einfügen einer Task vom Typ 'SYSTEM' in den Unterbaum 'Hardwarestruktur' des Projektbaums.
- Aufrufen des Dialogs 'Task ... einrichten', um das SPG auszuwählen, dem Sie die Funktionalität hinzufügen wollen.
- Zuweisen des Programms zur Systemtask.

4.4.3 Deklarieren von Anwender-Tasks

Um Anwender-Tasks in MULTIPROG zu deklarieren, führen Sie folgende grundlegende Schritte durch:

- Einfügen der Task im Unterbaum 'Hardwarestruktur' des Projektbaums.
- Auswählen der Taskeinstellungen im Dialog 'Task ... einrichten', den Sie über den Menüpunkt 'Einrichten...' aus dem Kontextmenü zur Task aufrufen können.
- Zuweisen von Programmen zu Tasks.

Sehen Sie zu detaillierten Informationen über das Einfügen von Tasks und Taskeinstellungen in der spezifischen Hilfe zu MULTIPROG nach.

4.4.4 Prioritäten der Anwender-Tasks

Die Tabelle gibt eine Übersicht über die empfohlenen Task-Prioritäten bzw. deren Einordnung hinsichtlich wichtiger reservierter Firmware-Tasks (tfwLAGE, tfwCANhigh, tfwIPO).

MULTIPROG-Priorität	RTOS*-Priorität (default)	RTOS* Task-Name	Verwendung
0	30	beliebig	z.B. Anwender–Task vor dem I/O-Refresh (Ereignis 8)
1	31	beliebig	PLC_CYCLE_RIOHIGH
2	30	beliebig	z.B. Anwender–Task nach dem I/O-Refresh (Ereignis 8)
4	34	beliebig	z.B. Anwender-Task (Ereignis 5)
5	35	tfwCANhigh	reserviert für CAN-Stack Task
6	36	beliebig	z.B. Anwender- Task (Ereignis 5)
7..15	41..45	beliebig	z.B. zyklische Anwender-Tasks
Default	127	default	Hintergrund-Task

*Real Time Operating System

Eine falsche oder zumindest ungeeignete Wahl der Anwender-Task-Einstellungen hinsichtlich Typ, Priorität oder Interrupt-Mode usw. - insbesondere in Kombination mit langen Programmlaufzeiten - kann zu Steuerungsfehlfunktionen führen, da essentielle Firmware-Tasks verdrängt werden.

Das System unterstützt insgesamt 18 Anwender-Tasks (Prioritätsstufen 0..16 und die Default-Task).
Tasks mit Prioritätswerten >= 16 werden mit Priorität 16 ausgeführt.

4.4.5 Anwender-Task-Informationen

Für jede Anwender-Task werden Informationen auf System-Variablen abgebildet. Die unten abgebildeten Typdefinitionen der Systemvariablen sind in der Bibliothek SchleicherLib im Abschnitt PLC_Types zu finden.

```
TYPE
 TaskNameType : ARRAY [1..10] OF BYTE;
END_TYPE

TYPE
 TaskInfoType0 : STRUCT
  MaxTask   : INT;      (* 00: *)          max. mögliche Taskanzahl
  CurTask   : INT;      (* 02: *)          aktuelle Taskanzahl
 END_STRUCT (* TaskInfoType0 *);
END_TYPE


TYPE
 TaskInfoType1 : STRUCT
  TaskName   : TaskNameType; (* 04: *)     Taskname
  TaskPrio   : INT;      (* 14: *)         Taskpriorität
  TaskMode   : INT;      (* 16: *)         Taskmode
  TaskPeriod  : INT;      (* 18: [ms] *)   Taskperiode in ms
  TaskStack  : INT;      (* 20: *)         Größe des benutzten Task-Stacks
  MainPoe   : INT;      (* 22: assigned PLC program *)  zugeordnetes SPS-Programm
  TaskWatchDog : INT;      (* 24: [ms] *)  Watch-Dog-Zeit in ms
  reserve0   : DINT;     (* 26: *)
  MaxStack   : INT;      (* 30: max. used stack *)  Größe des möglichen Task-Stacks
  CurDuration : INT;      (* 32: [ticks] *)  aktuelle Taskdauer einschließlich
                                             bevorrechtigte Aufrufe
  MinDuration : INT;      (* 34: [ticks] *)  minimale Taskdauer
  MaxDuration : INT;      (* 36: [ticks] *)  maximale Taskdauer
  AveDuration : INT;      (* 38: [ticks] *)  mittlere Taskdauer
  CurDelay   : INT;      (* 40: [ticks] *)  aktuelle Taskverzögerung
  MinDelay   : INT;      (* 42: [ticks] *)  minimale Taskverzögerung
  MaxDelay   : INT;      (* 44: [ticks] *)  maximale Taskverzögerung
  AveDelay   : INT;      (* 46: [ticks] *)  mittlere Taskverzögerung
 END_STRUCT (* TaskInfoType1 *);
END_TYPE
```

Die Variablen werden mit den Typen *TaskInfoType0* und *TaskInfoType1* deklariert.

Name	Typ	Verwendung	Beschreibung	Adresse	Anfangsw...	Reman...
TaskInfo0	TaskInfoType0	VAR_GLOBAL		%MD 1.1000		☐
TaskInfo1	TaskInfoType1	VAR_GLOBAL		%MD 1.1004		☐
TaskInfo2	TaskInfoType1	VAR_GLOBAL		%MD 1.1068		☐
TaskInfo3	TaskInfoType1	VAR_GLOBAL		%MD 1.1132		☐
TaskInfo4	TaskInfoType1	VAR_GLOBAL		%MD 1.1196		☐
TaskInfo5	TaskInfoType1	VAR_GLOBAL		%MD 1.1260		☐

Die folgende Anwender-Task-Information wird mit einem Offset von 64 ab 1004 deklariert (1004 + 64 = 1068 usw.).

Die Reihenfolge der Tasks wird durch den Rang der Task im Projektbaum *Physikalische Hardware/Configuration/Resource/Tasks* festgelegt.

4.4.6 Tasks und Watchdogs

Es gibt zu jeder anwenderdefinierten Task einen eigenen einstellbaren Watchdog.
Der Watchdog überprüft, ob die Taskausführung am Ende des Watchdog-Zeitintervalls beendet ist. Wenn die Taskausführung nach dieser Zeit nicht beendet wird, wird die Systemtask SPG 10 'WATCHDOG' ausgeführt und die SPS geht in den 'STOP'-Zustand über, wenn keine weiteren Aktionen programmiert wurden. Zusätzlich wird ein Eintrag in den Fehlerkatalog vorgenommen. Das Watchdog-Zeitintervall beginnt, wenn die Task bereit für die Ausführung ist. Das Watchdog-Zeitintervall wird im Dialog 'Task ... einrichten' in MULTIPROG festgelegt.

Wenn die Ausführungsdauer der Task sowie die Watchdog-Zeit annähernd denselben Wert haben und eine hohe CPU-Auslastung vorliegt, ist es möglich, dass während der Umsetzung einiger Online-Bedienschritte die Watchdog-Zeit überschritten wird.

Ein Grund für dieses Verhalten kann sein, dass Sie während des Debuggens im Online-Modus den Adressstatus mit Adressstatus ausgewählt haben.

Empfehlung:
Watchdog-Zeit für die Inbetriebnahme auf „0“ stellen, damit ist die Watchdog-Überwachung deaktiviert.

Online-Bedienschritte, die einen Lese- oder Schreibzugriff auf den Flash-Speicher verursachen, können die Ausführung der Task für kurze Zeit unterbrechen. Diese Online-Bedienschritte sind z.B.:

- Aufruf des Dialogs 'Ressource:Ressourcename' per Doppelklick mit der linken Maustaste auf die Schaltfläche 'Info' im Kontrolldialog
- Bootprojekt senden
- Archivprojekt senden

Die folgende Abbildung zeigt, dass die Task bei 10ms ihre Watchdog-Zeit überschreitet, da die Ausführung der Task unterbrochen wird.

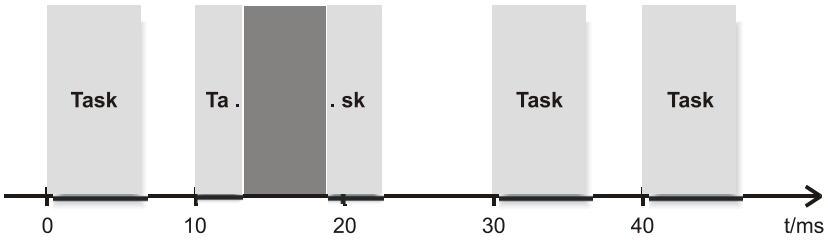


Abbildung 14: Watchdog

In diesem Beispiel ist die Watchdog-Zeit der angezeigten Task auf 10 ms eingestellt. In der Abbildung überschreitet die Task ihre Watchdog-Zeit nach 20 ms. Wenn die Watchdog-Zeit der Task auf 20 ms eingestellt ist, wird sie beim nächsten Mal nach 30 ms ausgeführt. In diesem Fall wird die Ausführung einer Task nach 20 ms übersprungen.

4.4.7 Tasks und Programme

Programme sind Programm-Organisationseinheiten oder POEs, wie in der IEC 61131-3 definiert. Die Programmcodes für Programme werden in MULTIPROG editiert.
Innerhalb von Programmen können weitere Programme, Funktionsbausteine oder Funktionen aufgerufen werden.

Programme müssen Tasks zugewiesen werden. Zuweisen eines Programms zu einer Task bedeutet, dass das Programm ausgeführt wird, wenn die Task aktiviert wird. In ProConOS und MULTIPROG können mehrere Programme einer Task zugewiesen werden.
In diesem Fall wird das erste Programm im Taskverzeichnis als erstes ausgeführt. Danach wird das Programm darunter ausgeführt usw.

4.5 SPS-Betriebszustände und Startverhalten

4.5.1 Betriebszustände

Betriebszustand	Beschreibung
EIN	• Stromzufuhr ist eingeschaltet • es ist kein Programm geladen
STOP	• Programm ist geladen • Anwender-Tasks sind inaktiv • Eingänge des Prozessabbildspeichers werden nicht aktualisiert • Ausgangssignale werden nicht an die Ein- und Ausgänge übermittelt
BETRIEB	• Programmausführung ist aktiviert • Anwender-Tasks sind aktiv • Eingänge des Prozessabbildspeichers werden gemäß der I/O-Konfiguration aktualisiert • Ausgänge des Prozessabbildspeichers werden gemäß der I/O-Konfiguration und der Programmausführung aktualisiert
HALT	• Programmausführung wird an einem Haltepunkt angehalten • Anwender-Tasks sind inaktiv • Eingänge des Prozessabbildspeichers werden nicht aktualisiert • Ausgänge des Prozessabbildspeichers werden nicht aktualisiert

Der aktuelle Zustand der SPS wird entweder oben im Kontrolldialog oder im Dialog 'Ressource: *Ressourcename*' in MULTIPROG angezeigt.
Wenn hinter dem aktuellen Zustand im Kontrolldialog 'Debug' angezeigt wird, bedeutet das, dass Haltepunkte gesetzt oder Variablen geforcet wurden.

Vor dem Ausführen einiger Funktionen, die Daten ins Anwenderflash schreiben, muss die Steuereinheit vom Anwender in den Betriebszustand STOP versetzt werden, da es sonst bei laufender Steuerung zu einem kurzen Abfall der Ausgänge kommen kann.

Tätigkeiten unter MULTIPROG, bei denen in STOP geschaltet werden muss:

- Bootprojekt senden
- Programmquelle senden
- Bootprojekt am Ziel löschen
- Programmquelle am Ziel löschen

Tätigkeiten in ProCANopen (mit Option „lokal speichern“), bei denen in STOP geschaltet werden muss:

- NMT-Konfigurationsdaten mit ProCANopen senden

4.5.2 Wechseln der Betriebszuständen mit MULTIPROG

Über die graphische Benutzeroberfläche von MULTIPROG kann gesteuert werden, wann die Programmausführung auf der SPS gestartet und gestoppt wird. Die Schaltflächen für Wechsel, die im aktuellen Betriebszustand nicht möglich sind, sind im Kontrolldialog abgeblendet.

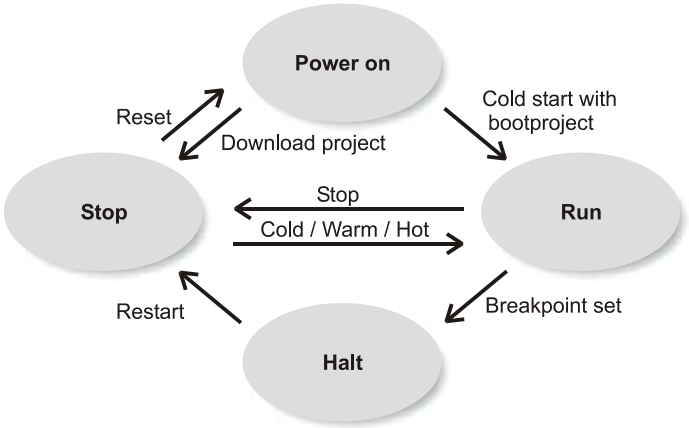


Abbildung 15: Betriebszustände

Starten der Programmausführung

Zustandswechsel von → nach	Schaltfläche im Kontrolldialog	Beschreibung, was passiert
Stop → Betrieb	Cold	• es erfolgt ein Kaltstart • alle Daten werden initialisiert • SPG 1 wird aufgerufen • alle Anwender-Tasks werden aktiviert • die Programmausführung wird aktiviert
Stop → Betrieb	Warm	• es erfolgt ein Warmstart • nur nicht-gepufferte Daten werden initialisiert • SPG 0 wird aufgerufen • alle Anwender-Tasks werden aktiviert • die Programmausführung wird aktiviert
Stop → Betrieb	Hot	• es erfolgt ein Heißstart • es werden keine Daten initialisiert • alle Anwender-Tasks werden aktiviert • die Programmausführung wird aktiviert • nicht verfügbar, wenn Sie die Programmausführung zum ersten Mal nach dem Senden starten

Stoppen der Programmausführung

Zustandswechsel von → nach	Schaltfläche im Kontrolldialog	Beschreibung, was passiert
Betrieb → Stop	Stop	• alle Anwender-Tasks werden deaktiviert, wenn ihr Arbeitszyklus beendet ist • SPG 2 wird aufgerufen • die Ausgänge des Prozessabbildspeichers werden geschrieben • die Programmausführung wird gestoppt • die physikalischen Ausgänge werden auf Null oder Vorzugsabschaltlage gesetzt

Allgemeines Reset

Zustandswechsel von → nach	Schaltfläche im Kontrolldialog	Beschreibung, was passiert
Stop → Ein	Reset	• das Projekt wird gelöscht • es erfolgt ein allgemeines Reset

4.5.3 Startverhalten der SPS nach dem Einschalten der Versorgungsspannung

Es können folgende Varianten ausgewählt werden:

- SPS-Stop (default)
- SPS-Warmstart nach IEC 61131-3
- SPS-Kaltstart nach IEC 61131-3

Das SPS-Startverhalten wird mit dem Betriebsartenschalter eingestellt.

4.6 SPS-Adressen

4.6.1 Diagnosedaten und Prozessabbild

Diagnosedaten	
Adresse	Inhalt
QB0 ... 3	Ausgangsdaten
QW0 ... 2	
QD0	
IB0 ... 3	Eingangsdaten
IW0 ... 2	
ID0	

Prozessabbild	
Adresse	Inhalt
IB4 ... 63	Eingangsdaten
IW4 ... 62	
ID4 ... 60	
IB64 ... 127	Eingangsdaten unverzögert
IW64 ... 126	
ID64 ... 124	
QB4 ... 63	Ausgangsdaten
QW4 ... 62	
QD4 ... 60	

Das unverzögerte Eingangsabbild muss einen eigen Eintrag in der sogenannten I/O-Konfiguration haben (siehe Kap. 4.7.2).

XFIO	Eingänge am XNT24 und der Betriebsartenschalter
Adresse	Inhalt
IX10000.0	Input ,0 am XNT24
IX10000.1	Input ,1 am XNT24
IB10002	Status des Betriebsartenschalters
IB10003	unbenutzt

4.6.2 Systemmerker

Systemmerker sind Merker, die über den Systemzustand informieren, wie z.B. über geforcte Variablen, Leistungsfähigkeit der CPU, etc. Diese Merker haben feste Speicheradressen und können vom SPS-Programm verwendet werden, um die entsprechenden Informationen zu erhalten.

Alle Systemmerker in der folgenden Tabelle sind bereits als globale Variablen im globalen Variablen-Arbeitsblatt deklariert, wenn Sie ein Projekt in MULTIPROG erstellen. Um sie in POEs zu verwenden, deklarieren Sie eine symbolische Variable im Variablen-Arbeitsblatt der POE, indem Sie das Schlüsselwort VAR_EXTERNAL und den Namen des Systemmerkers als Variablennamen verwenden.

Systemmerker

Name	Datentyp	Log. Adr. (Byte)	Log. Adr. (Bit)	Beschreibung
PLCMODE_ON	BOOL	0	0	TRUE := aktueller SPS-Zustand ist EIN
PLCMODE_RUN	BOOL	0	1	TRUE := aktueller SPS-Zustand ist BETRIEB
PLCMODE_STOP	BOOL	0	2	TRUE := aktueller SPS-Zustand ist STOP
PLCMODE_HALT	BOOL	0	3	TRUE := aktueller SPS-Zustand ist HALT
PLCDEBUG_BPSET	BOOL	1	4	TRUE := ein oder mehrere Haltepunkte sind gesetzt
PLCDEBUG_FORCE	BOOL	2	0	TRUE := eine oder mehrere Variablen sind geforct
PLCDEBUG_POWERFLOW	BOOL	2	3	TRUE := Adressstatus ist aktiv
PLC_TICKS_PER_SEC	INT	44	-	Anzahl der Systemticks pro Sekunde, die von der SPS als Systemzeitbasis verwendet werden. Dieser Wert bestimmt die Zeitauflösung der SPS für Funktionsbausteine für Zeitverzögerung, wie TON, TOF oder TP und die kürzeste Zykluszeit für die DEFAULT-Task und zyklische Tasks.
PLC_SYS_TICK_CNT	DINT	52	-	Anzahl der gezählten SPS Systemticks

Zusätzlich zu diesen Systemmerkern, die in der globalen Variablen-Deklaration zu sehen sind, können Sie für spezielle Zwecke Systemmerker verwenden, die Task- und Fehlerinformationen enthalten.

4.6.3 Feldbusspezifische SPS-Adressen

Auf den Netzvariablen werden die Daten abgelegt, die über das Netzwerk gesendet und empfangen werden.

Netzvariablen CANopen (siehe ProConOS IO-Treiber *CANIO*)	
Adresse	Inhalt
%IB1000 ... 1255	Eingangsdaten (BYTE)
	Eingangsdaten (WORD)
%ID1000 ... 1252	Eingangsdaten (DWORD)
%QB1000 ... 1255	Ausgangsdaten (BYTE)
%QW1000 ... 1254	Ausgangsdaten (WORD)
%QD1000 ... 1252	Ausgangsdaten (DWORD)

Netzvariablen PROFIBUS-DP (siehe ProConOS IO-Treiber *XDPIO*)	
Adresse	Inhalt
%IB5000 ... 5243	Eingangsdaten (BYTE)
%IW5000 ... 5242	Eingangsdaten (WORD)
%ID5000 ... 5240	Eingangsdaten (DWORD)
%QB5000 ... 5243	Ausgangsdaten (BYTE)
%QW5000 ... 5242	Ausgangsdaten (WORD)
%QD5000 ... 5240	Ausgangsdaten (DWORD)

Die Anzahl der verwendeten Netzvariablen sollte immer so gering wie möglich sein, um den Datenaustausch für die Kommunikation klein zu halten.

4.7 Konfigurieren der SPS

4.7.1 Eingänge, Ausgänge und der Prozess

Während der angeschlossene Prozess abläuft, erhält die SPS-Eingangssignale über die I/O-Module. Die Eingangssignale werden gemäß dem SPS-Programm abgearbeitet. Nach dem Arbeitszyklus überträgt die SPS die Ausgangssignale an die I/O-Module. Es wird durch den Anwender über die I/O-Konfiguration in MULTIPROG festgelegt, wie die SPS diese Signale verarbeitet.

In diesem Kapitel werden die Grundprinzipien einer I/O-Konfiguration erklärt.

4.7.2 Editieren der I/O-Konfiguration in MULTIPROG

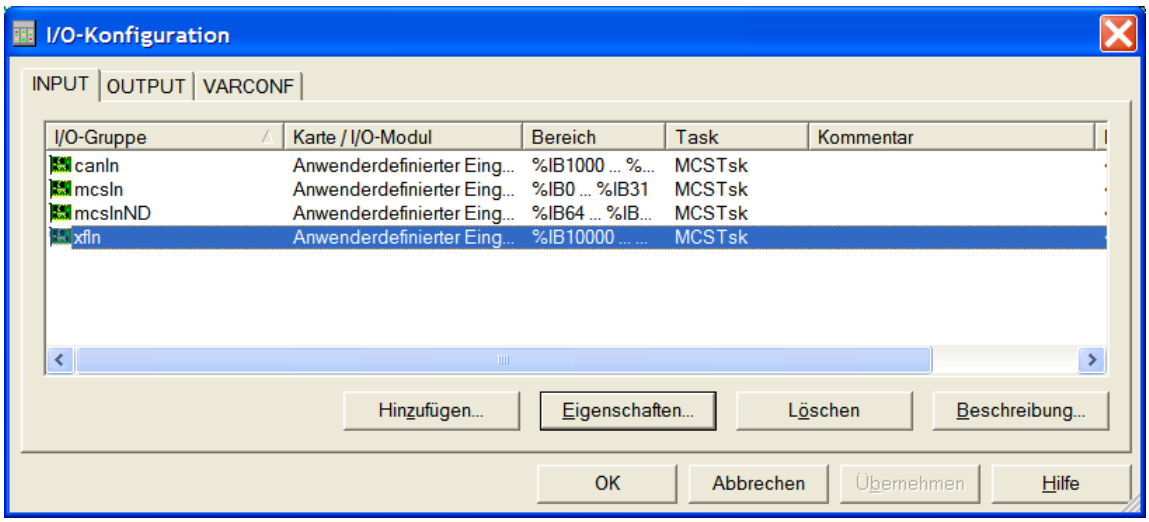
Eine I/O-Konfiguration definiert die I/O- und Netzwerkvariablen-Adressen, die im SPS-Projekt für eine Ressource verwendet werden.

Die I/O-Konfiguration in MULTIPROG wird mit dem I/O-Konfigurationseditor editiert. Um diesen Editor aufzurufen, doppelklicken Sie mit der linken Maustaste auf das Symbol 'IO_Configuration' im Unterbaum 'Hardwarestruktur' des Projektbaums.

Deklarieren der I/O-Gruppen

Im ersten Menü werden die I/O-Gruppe dargestellt. Je nach Ihrer Anwendungsaufgabe kann es sinnvoll sein, für jedes I/O-Modul eine I/O-Gruppe in MULTIPROG zu deklarieren.

Eine typische IO_Configuration ist im Beispiel dargestellt:



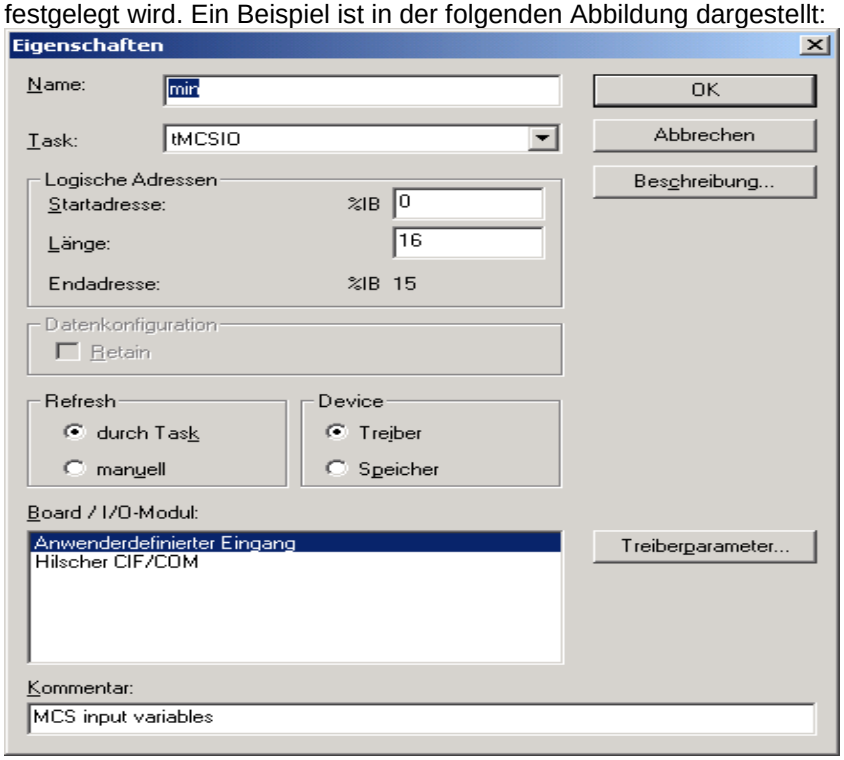
Der Modulname kann vom Anwender ausgewählt werden. Er bietet die Möglichkeit einer besseren Strukturierung der I/O-Konfiguration.

Sollen die unverzögerten Eingänge ab Adresse IX64.0 benutzt werden, muss ein eigener Eintrag in der I/O-Konfiguration vorgesehen werden. Für einen schnellen E/A Zugriff müssen die I/O Gruppen einer Task zugeordnet werden.

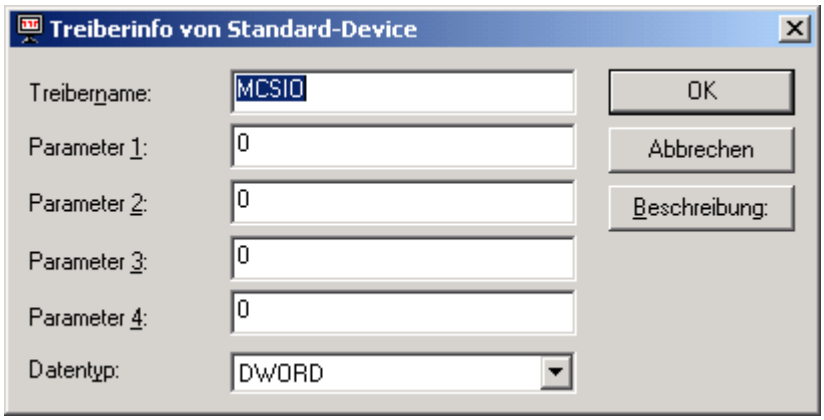
Für Eingangsmodule und Netzwerk-Eingangsvariablen muss der Modultyp 'INPUT' verwendet werden, für Ausgangsmodule und Netzwerk-Ausgangsvariablen der Modultyp 'OUTPUT'.

Deklarieren der logischen Adressen und der Zugriffsart

In den Eigenschaften der I/O-Gruppen werden die logischen Adressen und die Zugriffsart für die I/O-Gruppen deklariert.
Hier werden die E/A Adressen der I/O-Module bzw. die Netzwerk-E/A-Daten einem Speicherbereich im SPS-Speicher zugewiesen. Daneben wird die Art des Zugriffs (byte-, wort- oder doppelwortweise) festgelegt wird. Ein Beispiel ist in der folgenden Abbildung dargestellt:



In den Treiberparamtern steht , ob die Eingangs- und Ausgangsdaten byteweise, wortweise oder doppelwortweise gelesen bzw. geschrieben werden:



Wenn der Zugriff wortweise erfolgen soll, dann müssen die Anzahl der Eingangs- bzw. Ausgangsbytes sowie die Adresse des ersten Bytes gradzahlig sein.

Wenn der Zugriff doppelwortweise erfolgen soll, dann müssen die Anzahl der Eingangs- bzw. Ausgangsbytes sowie die Adresse des ersten Bytes ein Vielfaches von 4 sein.

Wenn die Adressen der Eingangs- bzw. Ausgangsbytes

- Analogmodulen,
- Zählermodulen,
- Temperaturmodulen oder
- Positioniermodulen

zugehörig sind, dann muss der Zugriff immer wortweise oder doppelwortweise definiert werden.

Die Treibernamen

Die Treibernamen für die verschiedenen Daten der Prozess-peripherie sind festgelegt:

Der Treibername für die Diagnosedaten und das Prozessabbild der I/O Module lautet ‘MCSIO‘.

Der Treibername für die schnellen Ein-/Ausgänge auf dem XNT24 lautet ‘XFIO‘.

Der Treibername für die Netzwerkvariablen von CANopen lautet ‘CANIO‘ (nur für Steuereinheit XCS20C).

Der Treibername für die Netzwerkvariablen von PROFIBUS-DP lautet ‘XDPIO‘ (nur für Steuereinheit XCS20P).

Editieren der Datenkonfiguration

Der Dialog 'Datenbereich' in MULTIPROG ermöglicht die Einstellungen der Datenbereiche. Dieser Dialog erscheint, wenn Sie auf die Schaltfläche 'Datenbereich...' im Dialog 'Ressourceeinstellungen...' klicken. Die Felder und Schaltflächen der Dialoge haben folgende Bedeutung:

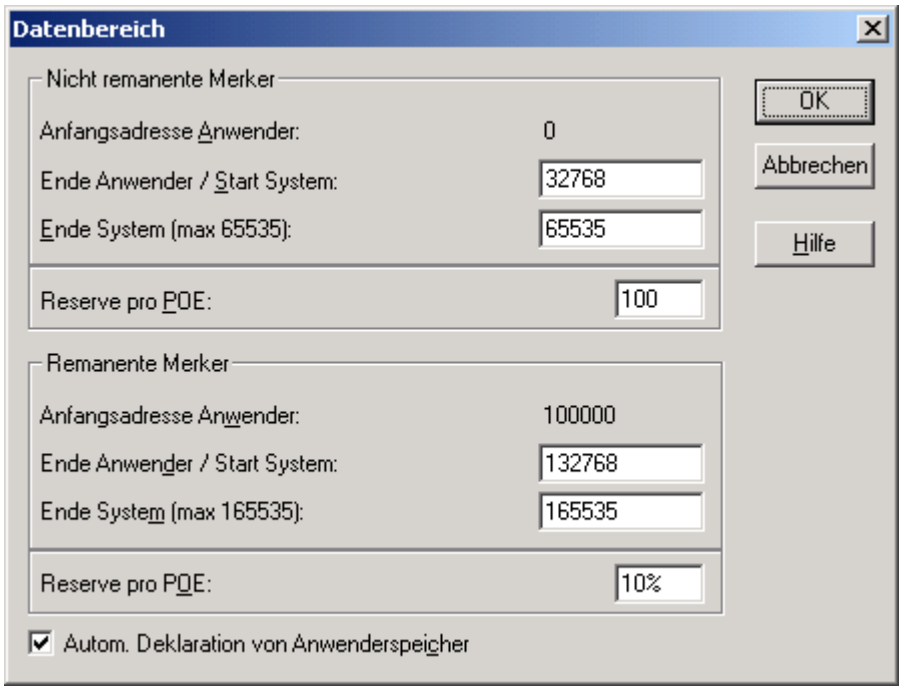
- Anfangsadresse Anwender: Gibt die Adresse an, wo der Speicherbereich für Anwendermerker beginnt.
- Ende Anwender / Start System: Gibt die Adresse an, wo der Speicherbereich für MULTIPROG-Merker beginnt.
- Ende System: Gibt die Adresse an, wo der Speicherbereich für MULTIPROG-Merker endet.
- Reserve pro POE: Gibt die Speicherreserve für eine POE an, die für Patch POE verwendet wird.
  Diese Reserve kann sowohl in % als auch in Bytes eingegeben werden.

MULTIPROG führt eine automatische Berechnung der Datenbereiche durch, die notwendig sind, um die MULTIPROG-Merker inklusive der Speicherreserve zu speichern.
Diese automatische Berechnung wird nicht bei Anwendermerker-Bereichen durchgeführt. Für weitere Informationen über die Projektgröße und den benötigten Speicherplatz für Merker klicken Sie auf das Register 'Infos' im Nachrichtenfenster, direkt nach dem kompilieren.

Bitte beachten Sie, dass MULTIPROG nicht überprüft, ob es eine Überlappung der logischen Adressen der Anwendermerker mit den MULTIPROG-Merkern gibt.

Während des Sendens kontrolliert die SPS alle Variablen sowie deren logische Adressen, die in einem SPS-Programm verwendet werden. Wenn ein Programm Variable logischen Adressen enthält, die nicht mit dem konfigurierten Datenbereich übereinstimmen, erscheint die folgende Fehlermeldung: 'Operand nicht implementiert oder Bereich überschritten bei...

Für die Konfiguration der Daten wird eine festgelegte Syntax verwendet. Die folgende Abbildung zeigt ein Beispiel für einen automatisch generierten verborgenen Eintrag von MULTIPROG:



Der Standardbereich für Speicheradressen von Anwendermerkern variiert je nach verwendeter Hardware.

4.8 Setzen und Auslesen der Echtzeituhr

Die Steuereinheit hat eine batteriegepufferte Echtzeituhr mit Kalender (Berücksichtigung von Schaltjahren) und einer Auflösung von 1 Sekunde.

Datum und Uhrzeit können mittels Funktionen aus der Bibliothek* Date_Time gelesen / gesetzt werden.

Ansteuerung der Uhr	
Funktion	
GET_TIME	Zeit lesen
GET_DATE	Datum lesen
SET_TIME	Zeit setzen
SET_DATE	Datum setzen

Die Laufzeit dieser Funktionen kann > 100 ms betragen.
Die Verwendung in einer langsamen zyklischen Task ohne Watchdog- Überwachung wird empfohlen.

4.9 Interrupt-Betrieb

Das Steuerungssystem XCx-micro stellt 2 schnelle digitale Eingänge (auf dem Netzteil-Modul XNT24) zur Verfügung.

I/O-Konfiguration

ProConOS-IO Treiber ***XFIO***
Adressen (max.) %IB 10000 ... 10003

```
(* Group:XFIO_Variables *)
VAR_GLOBAL
        xfio_in0        AT %IX 10000.0 :    BOOL; (* input 0 *)
        xfio_in1        AT %IX 10000.1 :    BOOL; (* input 1 *)
        xfio_sb0        AT %IB 10002 :      BYTE; (* state byte 0: key switch *)
        xfio_sb1        AT %IB 10003 :      BYTE; (* state byte 1: not used *)
END_VAR
```

Konfiguration

Die Konfiguration erfolgt mit Hilfe des Funktionsbausteins XFIO_CONFIG (FW-Lib PLC_Vxx, vgl. Hilfe).

Mit Hilfe von XFIO_CONFIG wird z.B. das Auslösen eines Interrupts (nur fallende Flanke ist möglich) an einem Interrupt-Eingang initiiert.

Außerdem kann der Interrupt-Modus wahlweise so eingestellt werden, dass nach dem Auftreten des Interrupts dieser automatisch deaktiviert wird ("Single Mode", MODE = 1).

In MWT sind hierbei die Ereignisse 0 und 1 den Interrupt-Eingängen X2- I0 bzw. I1 zugeordnet.

Beispiele

```
(* Channel #0 = Input 0, mode = 0 = continuos, falling edge *)
XFIO_CONFIG_1(ENABLE := enable1, CHANNEL := BYTE#0, MODE := BYTE#0,
                R_EDGE := FALSE, F_EDGE := FALSE);
error1    := XFIO_CONFIG_1.ERROR;
enable1   := FALSE;
```

```
(* Channel #0 = Input 0, mode = 1 = SINGLE, falling edge *)
XFIO_CONFIG_1(ENABLE :=enable2, CHANNEL := BYTE#0, MODE := BYTE#1,
                R_EDGE := TRUE, F_EDGE := TRUE);
error2    := XFIO_CONFIG_1.ERROR;
enable2   := FALSE;
```

```
(* Channel #1 = Input 0, mode = 0 = continuos, falling edge *)
XFIO_CONFIG_1(ENABLE := enable3, CHANNEL := BYTE#1, MODE := BYTE#0,
                R_EDGE := FALSE, F_EDGE := TRUE);
error3    := XFIO_CONFIG_1.ERROR;
enable3   := FALSE;
```

```
(* Channel #1 = Input 0, mode = 1 = SINGLE, falling edge *)
XFIO_CONFIG_1(ENABLE := enable4, CHANNEL := BYTE#1, MODE := BYTE#1,
                R_EDGE := FALSE, F_EDGE := TRUE);
error4    := XFIO_CONFIG_1.ERROR;
enable4   := FALSE;
```

4.10 Parametrier- und Diagnosefunktionen über die SPS

4.10.1 Übersicht

Die SPS kann in der Steuereinheit Parametrier- und Diagnosefunktionen auslösen.

Funktionen	
Code	Erklärung
0	Funktion 0 Sammelfehler auslesen
1	Funktion 1 Modul-Spannungsversorgung überwachen
2	Funktion 2 Überlast der Ausgangstreiber feststellen
3	Funktion 3 Prozessdatenbreite ermitteln
4	Funktion 4 Modul-Konfiguration auslesen
6, 7	reserviert
8	Funktion 8 Firmware-Version auslesen
9 ... 16	reserviert
17	Funktion 17 Busadresse auslesen
18	reserviert
19	reserviert
100 ... 110	reserviert
112, 113	Funktion 112 und 113 Baudraten der seriellen Schnittstellen einstellen
114	Funktion 114 Boot-Up-Delay einstellen
115	Funktion 115 SPS- Reboot
116	Funktion 116 CANopen-Task deaktivieren
117	Funktion 117 Zykluszeit für synchron gekoppelte Tasks einstellen
126	Funktion 126 Baudrate für CANopen einstellen
127	Funktion 127 Busadresse einstellen
255	Funktion 255 Reset

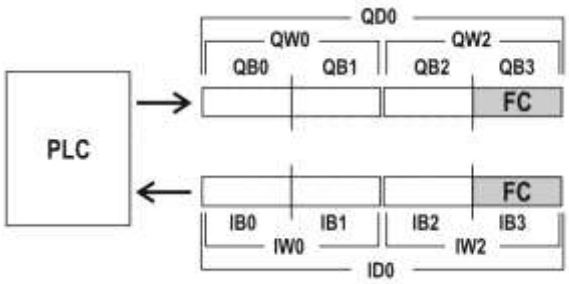
Die Steuereinheit führt, wenn angefordert, die Funktionen aus, bildet (wenn erforderlich) die Diagnosedaten und stellt diese der SPS zur Verfügung. Dort können die Daten im Anwenderprogramm ausgewertet und verarbeitet werden.

4.10.2 Adressierung und Datenaufbau

Die Parametrier- und Diagnosedaten werden auf den SPS-Adressen ab QD0 und ID0 abgebildet.

Parametrier- und Diagnosedaten	
Adresse	Inhalt
QB0, QB1, QB2, QB3	Ausgangsdaten
QW0, QW2	
QD0	
IB0, IB1, IB2, IB3	Eingangsdaten
IW0, IW2	
ID0	

Die QB3 und IB3 sind immer für den Funktionscodecode (FC) reserviert.



4.10.3 Ablauf

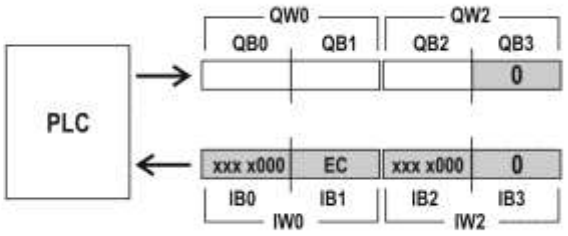
Die SPS fordert eine Funktion an, wenn in QB3 der Funktionscode eingetragen wird. Im QB0, QB1 und QB2 können erforderliche Parameter eingetragen werden.

Nach der Abarbeitung der Funktion wird als Quittung der Funktionscode im IB3 wiederholt.
Im IB0, IB1 und IB2 können dann Diagnosedaten gelesen werden.

Wird mehrmals aufeinanderfolgend die gleiche Funktion benutzt muss dazwischen die Funktion 255 (Reset) ausgeführt werden. Nur dann ist die richtige Auswertung der Diagnose gewährleistet.

4.10.4 Funktion 0 Sammelfehler auslesen

Diese Funktion liefert die Sammelfehlermeldung und den Fehlercode.



IB0 und IB2 Fehlerbits

In IB2 werden die Fehlerbit aus IB0 statisch gespeichert, bis die Funktion zurückgesetzt oder die Steuereinheit ausgeschaltet wird. Die Fehlerbits in beiden Bytes haben die gleiche Bedeutung.

Bit-Nr.	Bedeutung
0	Ein Modul des Busknoten ist überlastet.
1	Ein Modul des Busknoten hat einen 24 V-Fehler.
2	Eine Fehlermeldung wird generiert und an der Steuereinheit angezeigt. Die Fehlermeldung wird kodiert in IB1 übertragen.

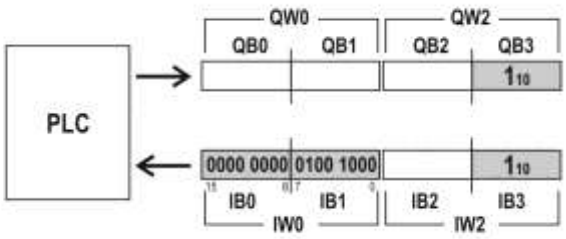
IB1 EC (Fehlercode)

Der Fehlercode ist gleich der Fehlermeldung, die an der Steuereinheit bei einigen Fehlern angezeigt wird.

Siehe dazu auch Seite 72

4.10.5 Funktion 1 Modul-Spannungsversorgung überwachen

Diese Funktion liefert den Zustand der 24 V-Spannungsversorgung der Module.



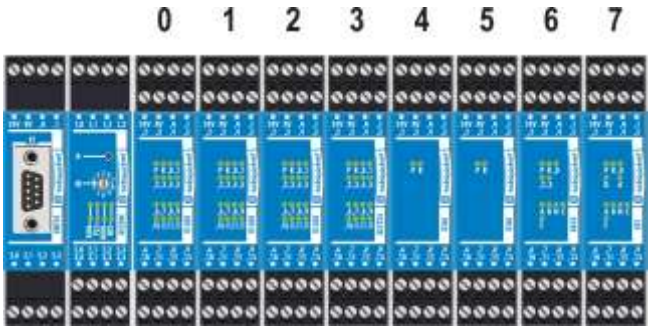
IW0 Fehlerbits

Bit-Nr.*	Bit-Wert	
0 ... 15	0	I/O-Modul wird korrekt mit 24 V versorgt
	1	I/O-Modul nicht mit 24 V versorgt

*Die Bit-Nr. entspricht der Modulnummer.

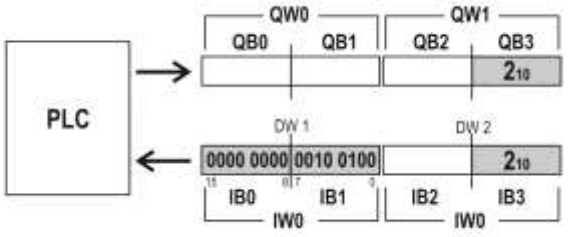
Im obigen Beispiel meldet die Steuereinheit für I/O-Modul 3 und 6 einen Fehler der 24 V-Spannungsversorgung.

Modulnummern (Platznummern):



4.10.6 Funktion 2 Überlast der Ausgangstreiber feststellen

Diese Funktion liefert den Zustand der 24 V-Ausgangstreiber.



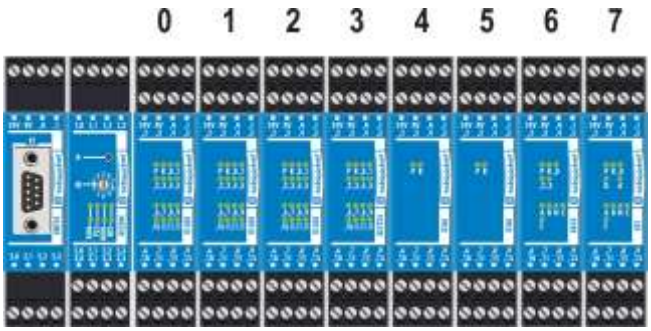
IW0 Fehlerbits

Bit-Nr.*	Bit-Wert	
0 ... 15	0	alle Ausgangstreiber des I/O-Moduls funktionieren korrekt
	1	mindestens einer der Ausgangstreiber des I/O-Moduls ist überlastet

*Die Bit-Nr. entspricht der Modulnummer.

Im obigen Beispiel meldet die Steuereinheit einen Überlast-Fehler der Ausgangstreiber für die I/O-Module 2 und 5.

Modulnummern (Platznummern):

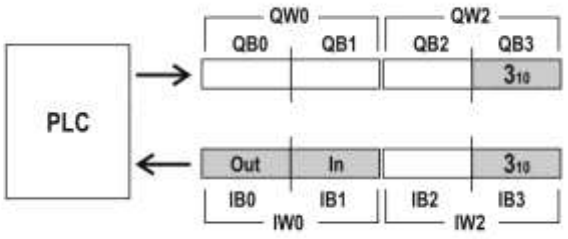


Diese Diagnose bezieht sich auf das gesamte Modul, nicht auf einzelne Kanäle.

Diese Diagnose ist nur gültig, wenn das betreffende Modul mit 24 V versorgt wird. Bei fehlender Ausgangstreiberversorgung ist die Diagnose "don´t care".

4.10.7 Funktion 3 Prozessdatenbreite ermitteln

Diese Funktion liefert die Prozessdatenbreite (Anzahl der E/A-Worte).

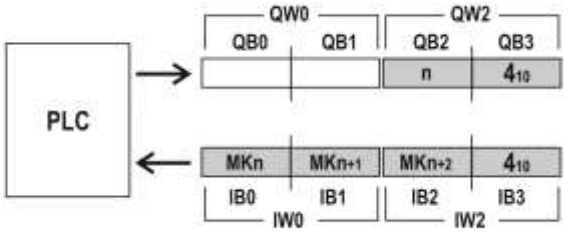


IW0 Out/In Anzahl der E/A-Worte

Byte	Bedeutung
IB0	Anzahl der Ausgangs-Bytes
IB1	Anzahl der Eingangs-Bytes

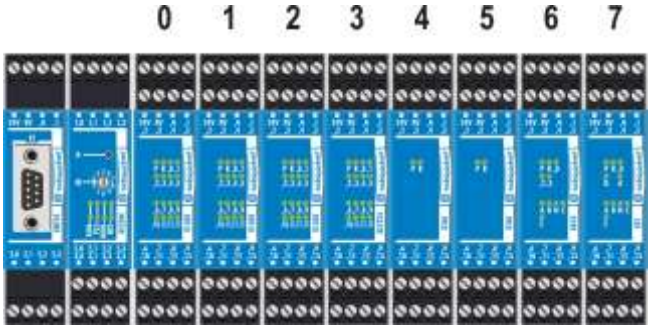
4.10.8 Funktion 4 Modul-Konfiguration auslesen

Diese Funktion liefert die Modulkennung von jeweils 3 I/O-Modulen.



In QB2 muss die Platznummer n eingetragen werden, ab der die Ist-Konfiguration auf den Plätzen n, n+1, n+2 ermittelt werden soll.

QB2 n



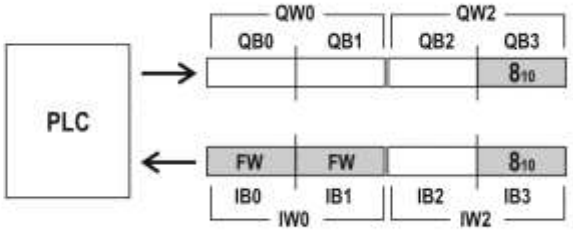
IB0, IB1, IB2 MK (Modulkennung*)

Byte	Bedeutung
IB0	Modulkennung des Moduls auf Platz n
IB1	Modulkennung des Moduls auf Platz n+1
IB2	Modulkennung des Moduls auf Platz n+2

*Die Modulkennungen sind in der Betriebsanleitung "RIO Erweiterungsmodule" Artikel-Nr. R4.322.1720.0 (322 154 14) zu finden.

4.10.9 Funktion 8 Firmware-Version auslesen

Diese Funktion liefert die Firmware-Version der Steuereinheit.



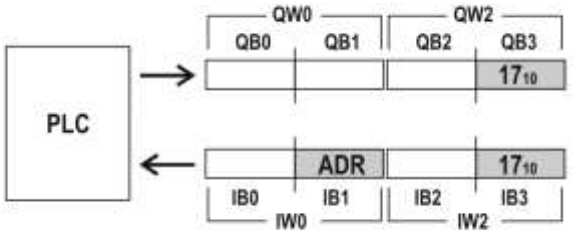
IW0 FW (Firmware-Version)

Die Firmware-Version wird hexadezimal-kodiert abgebildet.

4.10.10 Funktion 17 Busadresse auslesen

(Nur für Steuereinheiten mit Busfähigkeit)

Diese Funktion liefert die Busadresse.

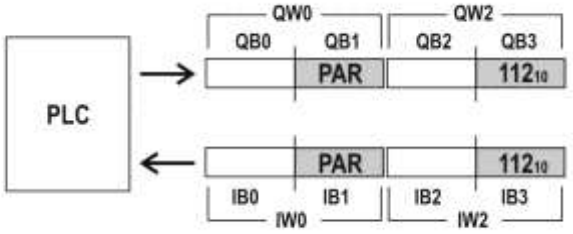


IB1 ADR

Busadresse der Steuereinheit

4.10.11 Funktion 112 und 113 Baudraten der seriellen Schnittstellen einstellen

Die Baudrate für die erste oder zweite serielle Schnittstelle können eingestellt werden.

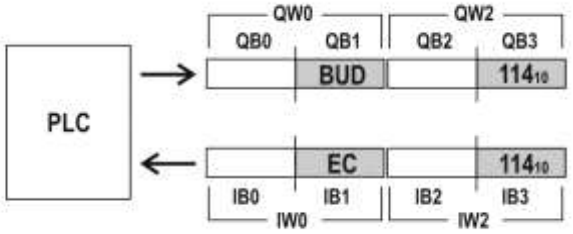


QB1 PAR (Baudrate)

Parameter 1	9600 Baud
Parameter 2	19200 Baud
Parameter 3	38400 Baud
Parameter 4	57600 Baud

Wird nach Power off-On übernommen.

4.10.12 Funktion 114 Boot-Up-Delay einstellen



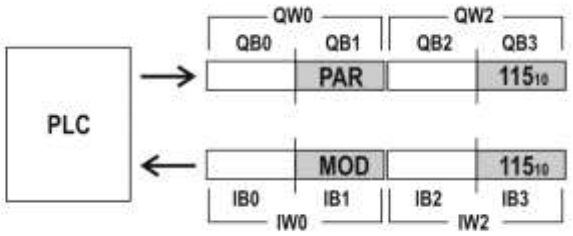
QB1 BUD (Boot-Up-Display)

0 ... 60	Boot-Up-Delay in Sekunden

IB1 EC (Fehlercode)

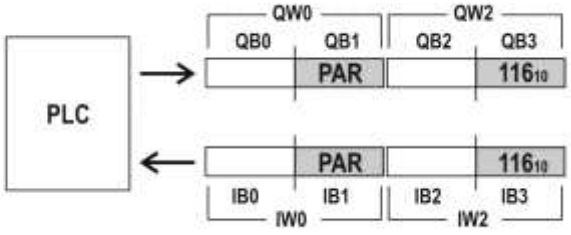
255	Fehler, Boot-Up-Delay außerhalb des Wertebereiches
1	kein Fehler

4.10.13 Funktion 115 SPS-Reboot



4.10.14 Funktion 116 CANopen-Task deaktivieren

Schaltet die CANopen-Task ein oder aus.

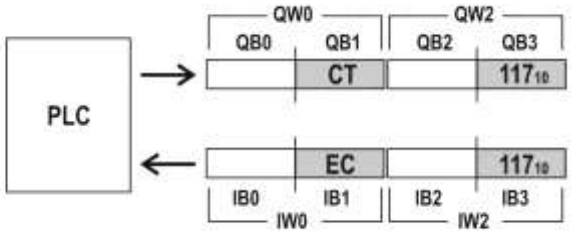


QB1 PAR

Parameter 0	Task deaktiviert
Parameter 1	Task aktiviert

Wird nach Power off-on üernommen.

4.10.15 Funktion 117 Zykluszeit für synchron gekoppelte Tasks einstellen



QB1 CT (Zykluszeit)

8 ... 20	Zykluszeit

IB1 EC (Fehlercode)

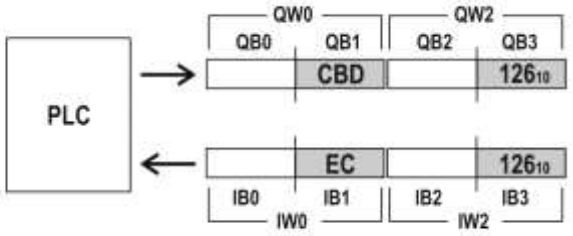
255	Fehler, außerhalb des Wertebereiches
1	kein Fehler

Synchron gekoppelte Tasks sind:
RIO-High-Level-Task, SPS-Task 5, CANopen-Task, PROFIBUS-Task
Die Defaulteinstellung ist 12 ms.

Die Einstellung ist nur in 4-ms Schritten möglich : 8 – 12 – 16 – 20 – 24

Wird nach Power off-on übernommen.

4.10.16 Funktion 126 Baudrate für CANopen einstellen



QB1 CBD (CANopen-Baudrate)

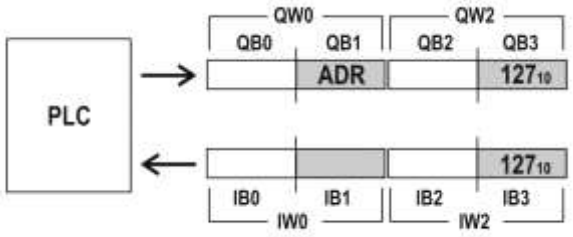
3	125 kBd
4	250 kBd
5	500 kBd
6	800 kBd
7	1000 kBd

IB1 EC (Fehlercode)

255	Fehler, außerhalb des Wertebereiches
1	kein Fehler

Wird nach Power off-on übernommen.

4.10.17 Funktion 127 Busadresse einstellen

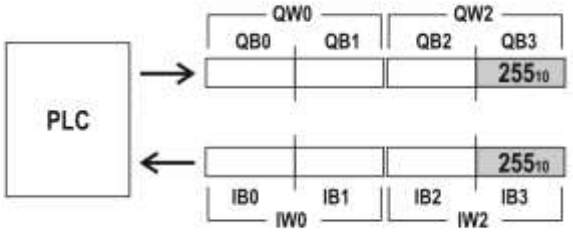


QB1 ADR
Busadresse der Steuereinheit

Wird nach Power off-on übernommen.

4.10.18 Funktion 255 Reset

Setzt die vorher ausgeführte Funktion zurück.



4.11 Interaktives Diagnose - Interface

4.11.1 Übersicht

Die unter Punkt 4.10 definierten PDF's können auch über ein interaktives Diagnose - Interface an die Steuerung gesendet werden. Um diese Möglichkeit nutzen zu können ist die Steuerung mit dem PC und einem dort gestarteten Terminalprogramm zu verbinden.

Als Terminalprogramm kann zum Beispiel das von Windows mitgelieferte Programm „Hyper Terminal" verwendet werden.

4.11.2 Einstellungen

Die folgenden Einstellungen beziehen sich auf das von Windows mitgelieferte Terminalprogramm „Hyper Terminal".

Starten Sie das Windows „Hyper Terminal" oder ein anderes Terminalprogramm und stellen Sie bitte folgende Parameter ein:

Schnittstellen und Terminal - Einstellungen

Emulation	TTY
Telnet Terminalkennung	ANSI
Bits pro Sekunde	38400
Datenbits	8
Parität	Keine
Stoppbits	1
Flusssteuerung	Kein
lokales Echo	Aus

Einstellung des Betriebsartenschalters

Der Betriebsartenschalter ist auf die Stellung 0 zu bringen. Dies kann auch bei eingeschalteter Steuerung erfolgen.

4.11.3 Eingabe der PDF - Codes

Wird die Steuerung nach dem Start des Terminalprogramms neu gestartet und befindet sich der Betriebsartenschalter auf der Stellung „0", werden im Terminalfenster einige Informationen während des Startvorgangs angezeigt.

Ist der Startvorgang beendet, startet die interaktive Eingabe der PDF – Codes mit der Ausgabe:

- Enter -

PDF-Code	*[0]* >	geben Sie hier den PDF-Code ein (z.B. 126 = Baudrate CAN-Schnittstelle) und betätigen Sie die Enter-Taste
PDF-Value	[0] >	geben Sie hier den Parameter für den PDF-Code ein (z.B. 3 = 500kbaud)

Als Ausgabe bekommen Sie dann folgende Anzeige, wobei sich die Definitionen „Output“ und Input aus Sicht des Terminalprogramms darstellen!

- Output -
PDF-Code = 126
PDF-Value = 3
Hex-Out = 0x0003007e
- Input -
PDF-Code = 126
PDF-Value = 3
Hex-In = 0x0003007e

Zu beachten:

Nach Eingabe eines PDF – Codes bleiben die Werte für den PDF – Code und den PDF – Value auf dem vorher eingegeben stehen und werden mit der Betätigung der Enter – Taste erneut übernommen.

4.12 Clear User - Flash

4.12.1 Beschreibung

Clear User - Flash

Über den Betriebsartenschalter ist die Möglichkeit gegeben, das User – Flash zu löschen bzw. neu zu formatieren.

Dazu ist wie folgt vorzugehen:

0. Stellung Test (0)

Betätigen des Reset – Schalters

Wenn die LED PLC-Status blinkt, umschalten auf

1. Stellung STOP (1)
2. Stellung Test (0)
3. Stellung WARM (9)
4. Stellung Test (0)

Erfolgt die Umschaltung innerhalb von 2 Sekunden, wird das Flash gelöscht und es erfolgt ein Reset

(Gelöscht wird das Boot-Projekt, das ZIP-Projekt und die CAN-Konfiguration)

4.13 Konformität gemäß IEC 61131-3

Die SPS ist weitgehend konform zur IEC 61131-3. In der allgemeinen Hilfe und im Handbuch für MULTIPROG befindet sich ein einleitendes Kapitel zur IEC 61131-3. Die IEC - Konformitätsliste finden Sie am Ende dieses Kapitels.

Einschränkungen für Datentypen

- Elementare Datentypen: LINT, LREAL, REAL, STRING, TIME-OF-DAY, DATE-AND-TIME und LWORD sind nicht implementiert.
- Anwenderdefinierte Datentypen können nur innerhalb von Funktionsbausteinen und Programmen verwendet werden. Sie können nicht innerhalb von Funktionen verwendet werden.
- Anwenderdefinierte Datentypen können nicht initialisiert werden.
- Anwenderdefinierte Datentypen können nicht geforct oder überschrieben werden. Ausnahme: im Watch-Fenster können Multielement-Datentypen geforct und überschrieben werden.
- Beim Debuggen von Feldern und Strukturen müssen die Variablen in das Watch-Fenster kopiert werden. Die Variablen können nur aus dem Variablen-Arbeitsblatt und nicht aus dem Programmcode-Arbeitsblatt in das Watch-Fenster kopiert werden.
- Alias-Datentypen und Datentypen für Unterbereiche sind nicht verfügbar.
- Multidimensionale Felder können nicht deklariert werden. Es ist jedoch möglich, Felder von Feldern zu deklarieren, wie im folgenden Beispiel dargestellt:

```
Datentyp-Deklaration:
TYPE
        graph           :       ARRAY [1..10] OF INT;
        mein_feld       :       ARRAY [1..3] OF graph;
END_TYPE

Variablen-Deklaration:
VAR
        var1    :       mein_feld;
        var2    :       INT;
END_VAR

Programmcode-Deklaration in ST:
var2    := var1[1] [3];
```

- Werden Variablen für Felder und Strukturen vom Datentyp BOOL an Eingängen oder Ausgängen verwenden, wird eine Fehlermeldung angezeigt.
- Als Feldindex kann nur der Datentyp 'INT' verwendet werden. Ausdrücke wie 'i + 1' sind nicht möglich.
- In Feldern entspricht der Datentyp BOOL einem Byte.
- Wenn adressierte Variablen für Felder und Strukturen verwendet werden, muss der Anwendungsprogrammierer überprüfen, ob die verwendeten logischen Adressen wirklich vorhanden sind. In MULTIPROG wird dies nicht überprüft.
- Felder in einer Deklaration einer Struktur müssen wie im folgenden Beispiel deklariert werden.

```
TYPE
      perform_def       :       ARRAY[1..23] OF INT;
      maschine : STRUCT
                  x_pos    :       INT;
                  leistung :       perform_def;
      END_STRUCT;
END_TYPE
```

Folgende Deklaration ist falsch!

```
TYPE
      maschine : STRUCT
                  x_pos    :       INT;
                  leistung :       ARRAY[1..23] OF INT;
      END_STRUCT;
END_TYPE
```

Einschränkungen für ST

- In IF-Anweisungen können boolesche Variablen verglichen werden, wie im folgenden Beispiel dargestellt (var1 ist eine boolesche Variable): IF var1=TRUE THEN ...;
  Es empfiehlt sich jedoch, eine andere Darstellungsweise zu verwenden, da dies schneller in der Ausführung und beim Kompilieren ist und da weniger Code erzeugt wird.
  Folgende Darstellungsweise verwenden: IF var1 THEN ....;
- In einer FOR-Anweisung kann nur aufwärts gezählt werden. Daher muss der Inkrementwert hinter dem Schlüsselwort BY immer positiv sein.
- In einer FOR-Anweisung können die Variable 'a' und der Inkrementwert innerhalb des Codes der Anweisung geändert werden. Es wird keine Fehlermeldung oder Warnung angezeigt, wenn das folgende Beispiel eingeben wird:
  FOR i=0 TO 10 BY5
  DO i:=10;
  Solche Anweisungsstrukturen äußerst vorsichtig verwenden.
- In einem Funktionsaufruf können keine Zuweisungen verwendet werden. Das folgende Beispiel ist nicht möglich: a:= SIN(b:=3);
- In einem Ausdruck werden alle Operationen verarbeitet. Teile eines Ausdrucks, die keine Auswirkung auf das Ergebnis haben, werden ebenfalls verarbeitet. Im folgenden Beispiel wird c>d auch verarbeitet, wenn a<b FALSE ist.
  Beispiel: a:=a<b AND c>d;

5 Feldbus CANopen

5.1 Grundlagen

CANopen basiert auf dem CAN Application Layer für industrielle Anwendungen CAL. Das CANopen-Kommunikationsprofil CiA DS-301 spezifiziert die Mechanismen zur Konfiguration und Kommunikation zwischen Geräten in Echtzeitumgebungen. CANopen benutzt die Datenübertragungschicht nach ISO 11898 und CAN 2.0 A+B.

- Bis zu 64 Teilnehmer an einem Bus möglich
- Beschreibung der Gerätedetails über ein EDS (Electronic Data Sheet)
- Objektorientierte Kommunikation mit PDOs und SDOs
- Übertragung von Echtzeitdaten mit ´purem´ CAN als PDO (Process Data Object)
- Komplexe oder niederpriore Dienste werden mit SDO (Service Data Object) übertragen
- PDOs können von allen Slaves ereignisgesteuert oder synchronisiert gesendet werden
- CANopen-Master übernehmen z.B. das Netzwerkmanagement bei Netzanlauf, sind aber nicht zur Kommunikation der Slaves untereinander notwendig

5.1.1 Belegung der Prozessdatenobjekte (PDO)

PDOs werden generiert, wenn sich die Eingangssignale geändert haben.

5.1.2 Nodeguarding

Über das Nodeguarding kann ein Master den Ausfall eines Slaves erkennen. Dazu sendet er zyklisch Nachrichten auf den Guarding-Identifier (100Eh) des Slaves. Dieser antwortet mit einer Guarding-Nachricht, die u.a. ein Toggle-Bit enthält.

5.1.3 Lifeguarding

Während das Nodeguarding vom Master durchgeführt wird benutzt der Slave diese Guarding-Telegramme, um seinerseits den Ausfall des Masters zu erkennen. Diese Überwachungsfunktion des Slaves wird Lifeguarding genannt.

Eine Kabelbrucherkennung und damit eine Zwangsabschaltung der Ausgänge kann bei CANopen nur bei aktiviertem Node- und Lifeguarding erfolgen !

Zur Aktivierung des Lifeguardings muss der Master die Objekte Guard-Time (100Ch) und den Life-Time-Factor (100Dh) beschreiben. Ist eines der beiden Objekte gleich 0, wird kein Lifeguarding und damit auch keine Kabelbrucherkennung durchgeführt !

Die Steuereinheit aktiviert den Fehler 6 wenn die Überwachungszeit abläuft, ohne dass ein Guarding-Telegramm beim Slave eintrifft. Der Fehler 6 bewirkt eine Zwangsabschaltung aller Ausgänge.

Die Überwachungszeit wird nach folgender Formel Berechnet.

*Life-Time = Life-Time-Factor * Guard-Time [ms]*

5.2 CANopen-Konfigurationsdaten

5.2.1 EDS-Dateien

DieDateien für alle Schleicher-Geräte können kostenlos vom Internet http://www.schleicher-electronic.com geladen werden.

5.3 Verkabelung CANopen

Kabellängen

Die Kabellänge ist abhängig von der verwendeten Datenübertragungsrate.

Kabellängen	
Datenübertragungsrate in kBaud	Kabellänge in m
125	500
250	250
500	100
800	50
1000	30

5.3.1 Abschlusswiderstände CANopen

An beiden Enden der Fernbusleitung muss jeweils ein Abschlusswiderstand von 120Ω zwischen CAN_L und CAN_H angebracht werden.

5.4 Einstellen der Knotennummern CANopen

Knotennummern müssen eindeutig festgelegt werden, doppelte Vergabe von Knotennummern führt zu Fehlern, die die Inbetriebnahme des Netzes verhindern. Es können Nummern im Bereich 1 bis 127 vergeben werden. Die Knotennummer 127 wird von der Projektierungssoftware ProCANopen belegt.

Die Knotennummern (Busadresse) werden über die Paramatrier- und Diagnosefunktion 127 eingestellt (siehe Seite 55).

5.5 Einstellen der Datenübertragungsrate CANopen

Mit der Parametrierfunktion 126 wird der Parameter eingestellt.

Parameter	Baudrate in kBaud
3	125
4	250
5	500
6	800
7	1000

6 Feldbus PROFIBUS-DP

PROFIBUS wurde 1983 als offener Feldbus entwickelt, 1991 in DIN 19245 genormt und ist seit 1996 mit der EN 50170 ein europäischer Standard.
PROFIBUS-DP ist speziell für Fertigungsautomatisierung mit dezentraler Peripherie ausgelegt.

Beim Planen einer Anlage sind neben den örtlichen/baulichen Gegebenheiten, die im wesentlichen die Standorte der Maschinen und Feldgeräte bestimmen, auch die physikalischen Vorschriften einer PROFIBUS-Anlage nach EN 50170 einzuhalten.

Die feldgerätespezifischen Installationsanweisungen der unterschiedlichen Lieferfirmen, sowie die sicherheitstechnischen Richtlinien einer Anlage haben weiterhin Gültigkeit.

6.1 Bustopologie PROFIBUS-DP

Gemäß PROFIBUS-RS485-Spezifikation können maximal 32 Teilnehmer an einem Bussegment angeschlossen werden. Um eine größere Anzahl an PROFIBUS-DP-Teilnehmern betreiben zu können, muss die Anlage durch Repeater segmentiert werden.

Repeater verbinden Bussegmente elektrisch miteinander und sorgen für die Verstärkung/Signalauffrischung der Datensignale. Repeater können zusätzlich zur galvanischen Trennung von Bussegmenten oder Bus-Teilabschnitten eingesetzt werden. Mit jedem Einsatz eines Repeaters kann ein PROFIBUS-System um ein weiteres Bussegment mit voller Leitungslänge und den maximal anschließbaren Feldgeräten erweitert werden. Durch den Einsatz von Repeatern treten Signalverzögerungen auf. Bei der Projektierung ist dies zu berücksichtigen.

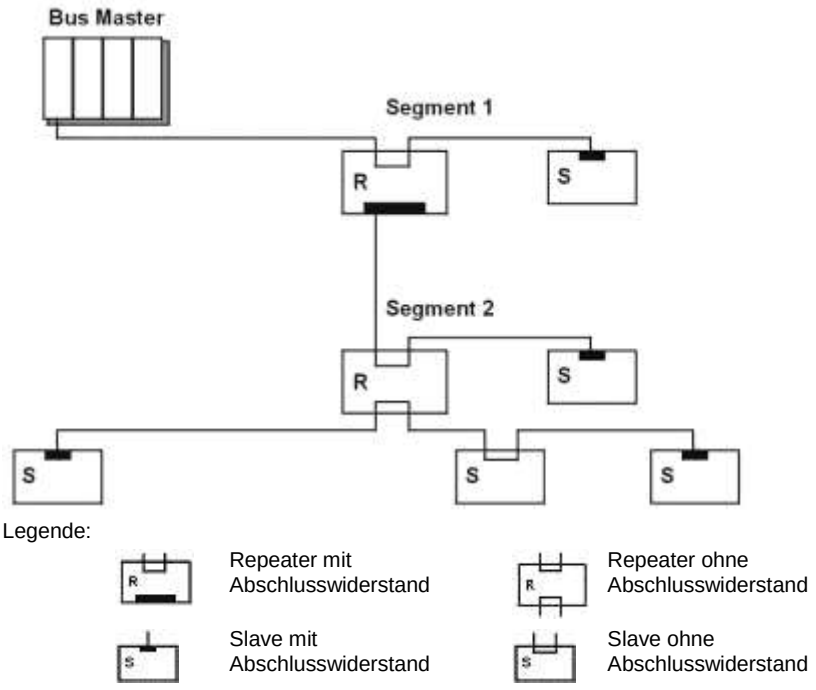


Abbildung 16: Bustopologie PROFIBUS-DP

Max. Anzahl der Teilnehmer im Vollausbau	126 (Adressen von 0 ... 125)
Anzahl Teilnehmer pro Segment inkl. Repeater	32
Übertragungsraten (die dem Master vorgegebene Übertragungsrate wird dem Slave mitgeteilt)	9.6, 19.2, 45.45, 93.75, 187.5, 500, 1500, 3000, 6000, 12 000 kBit/s
Anzahl der Segmente in Reihe (abhängig von den eingesetzten Repeatern und den eingestellten Busparametern)	3

Der Ausfall oder Abschalten einzelner Slaves während des laufenden Busbetriebs ist möglich. Andere Slaves können weiter betrieben werden.

Die komplette Bustopologie ist im Master projektiert.

Jeder Slave besitzt eine herstellerspezifische Identnummer, die durch die PROFIBUS Nutzerorganisation (PNO) vergeben wird.

6.2 Aufbaurichtlinien für PROFIBUS-Netze

Wichtige Maßnahmen zur Leitungsführung und Inbetriebnahme von PROFIBUS-Netzen werden in den Aufbaurichtlinien für PROFIBUS-Netze dargestellt. Herausgeber der Aufbaurichtlinien ist die PROFIBUS-Nutzerorganistation (PNO).

Zusätzlich, zu den in dieser Betriebsanleitung gemachten Angaben, ist die Aufbaurichtlinie der PROFIBUS-Nutzerorganistation (PNO) zu beachten.

Die Aufbaurichtlinie kann unter der Bestell-Nr. 2.111 bezogen werden von der:

PROFIBUS-Nutzerorganisation e. V.
Haid-und-Neu-Straße 7
76131 Karlsruhe

Telefon: +49 (0)721 / 96 58 590

Fax: +49 (0)721 / 96 58 589

www.profibus.com

PROFIBUS_International@compuserve.com

Die folgenden Aufbaurichtlinien beziehen sich ausschließlich auf die Übertragung mit Kupferleitungen (RS 485) gemäß EN 50170.

Die im Kapitel "Elektrische Installation" aufgeführten Richtlinien sind zusätzlich für alle Steuereinheiten gültig und müssen beachtet werden.

6.3 Buskabel PROFIBUS-DP

Die Eigenschaften der Busleitung sind in der EN 50170 Part 8-2 als Leitungstyp A spezifiziert.

Parameter	Wert
Wellenwiderstand (Ω)	135 ...165 (bei einer Frequenz von 3...20 MHz)
Kapazitätsbelag (pF/m)	< 30
Schleifenwiderstand (Ω/km)	<= 110
Aderndurchmesser (mm)	> 0,64*
Adernquerschnitt (mm^2)	> 0,34*

* Die verwendeten Adernquerschnitte müssen den Anschlussmöglichkeiten am Busstecker entsprechen.

6.4 Bussegmentlänge PROFIBUS-DP

Datenübertragunsrate in kbit/s	max. Bussegmentlänge in m
9,6	1200
19,2	1200
93,75	1200
187,5	1000
500	400
1500	200
12000	100

Innerhalb einer PROFIBUS-DP-Anlage kann nur eine Datenübertragungsrate gewählt werden.

6.5 Busverkabelung PROFIBUS-DP

Die beiden Datenleitungen werden bei PROFIBUS auch mit A und B bezeichnet. Es gibt keine Vorschrift, welche Datenleitungsadernfarbe an welche Klemme anzuschließen ist, sie muss nur innerhalb der gesamten Anlage (über mehrere Teilnehmer und Segmente hinweg) einheitlich sein.

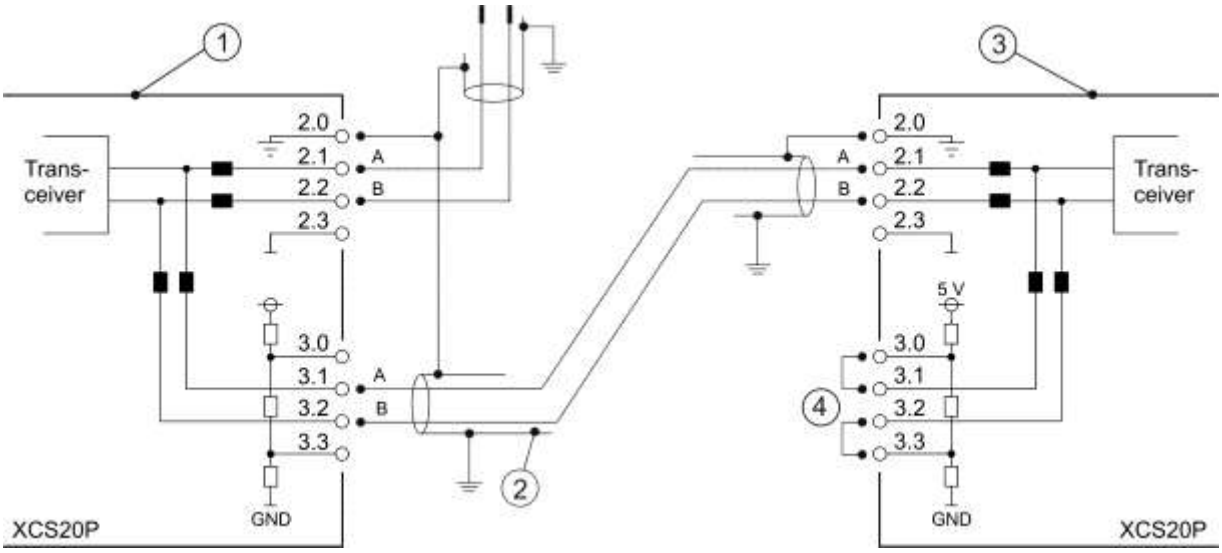
Wenn ein Übertragungskabel mit den Datenleitungsadern rot und grün verwendet wird, sollten folgende Zuordnungen verwendet werden:

Datenleitungsader A - grün

Datenleitungsader B - rot

Die Bezeichnungen gelten sowohl für die ankommende und abgehende Datenleitungsader.

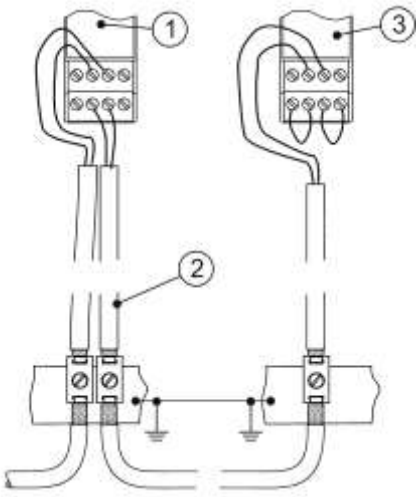
Busknoten und Busabschlüsse sind wie folgt zu verdrahten:



1 PROFIBUS-Knoten
2 abgeschirmtes Buskabel
3 PROFIBUS-Abschluss
4 Abschlusswiderstände müssen aktiviert werden

Abbildung 17: Busverkabelung PROFIBUS-DP

Die Schirme der Buskabel müssen an der Schrankeinführung großflächig und gut leitend auf die Potentialausgleichsschiene aufgelegt werden. Die Potentialausgleichsschiene ist bei jedem Elektronikschrank geerdet und mit den Potentialausgleichsschienen anderer Schränke verbunden.



1	PROFIBUS-Knoten
2	abgeschirmtes Buskabel
3	PROFIBUS-Abschluss

Abbildung 18: Busverkabelung (Schirmung) PROFIBUS-DP

Die im Kapitel "Elektrische Installation" aufgeführten Richtlinien sind zusätzlich für alle Steuereinheiten gültig und müssen beachtet werden.

6.6 Projektierung PROFIBUS-DP

Folgende Punkte sind bei der Projektierung durchzuführen:

- GSD-Datei mittels Konfigurator oder Programmiersystem einlesen.
- PROFIBUS-DP Mastersystem konfigurieren,
  Baudrate, höchste L2 Adresse etc. bestimmen,
  Busadresse des PROFIBUS-DP Masters festlegen.
- I/O-Ausbau des Busknoten projektieren und
  Busadresse festlegen.
- Ein- bzw. Ausgangsadresse des Busknoten festlegen.
- Die festgelegte Adresse des Busknoten am Buskoppler einstellen.
- Konfiguration in den PROFIBUS-DP Master übertragen.
- PROFIBUS-DP Master Steuerung programmieren,
  PROFIBUS-DP Eingangsdaten lesen,
  PROFIBUS-DP Ausgangsdaten schreiben.
- System in Betrieb setzen.

Weitere Ausführungen, speziell für die Inbetriebnahme mit STEP7, befinden sich in der Beschreibung "Inbetriebnahmehinweise für Feldbussysteme".

Es ist möglich die GSD-Datei vom Internet *http://www.schleicher-electronic.com* kostenlos zu laden.

GSD-Datei

schl0A57.gsd (GSD_Revision 1)

Modulkonfiguration

Jede beliebige (Modul)-Konfiguration (mit und ohne Konsistenz) in den gegebenen Maximalgrenzen kann durch die Master definiert werden:

Max_Input_Len = 244

Max_Output_Len = 244

ist konfigurierbar.

6.7 Inbetriebnahme PROFIBUS-DP

Siehe dazu Betriebsanleitung "Inbetriebnahmehinweise für Feldbussysteme".
Alle Betriebsanleitungen können kostenlos vom Internet http://www.schleicher-electronic.com geladen werden.

6.8 Diagnose am PROFIBUS-DP

In den Octets* 1 bis 6 stellt der Buskoppler die PROFIBUS-DP Standarddiagnose zur Verfügung.
Siehe auch DIN 19245 Teil 3 S. 40 ff.

(*) In der DIN 19245 wird ein Byte als Octet bezeichnet. Diese Bezeichnung wird auch hier verwendet.

Octet	Bit	Kurzbezeichnung	Beschreibung
1	0	non_exist	Slave existiert nicht (setzt Master)
	1	station_not_ready	Slave nicht für den Datenaustausch bereit
	2	cfg_fault	Konfigurationsdaten stimmen zwischen Master und Slave nicht überein
	3	ext_diag	es existieren erweiterte Diagnosebytes
	4		
	5	invalid_slave_response	vom Slave immer auf 0 gesetzt
	6	prm_fault	fehlerhafte Parametrierung
	7	master_lock	Slave ist von einem Master parametriert
2	0	prm_req	Slave muss neu parametriert werden
	1	stat_diag	statische Diagnose
	2		immer 1
	3	wd_on	Ansprechüberwachung aktiv
	4	freeze_mode	Freeze Kommando aktiv
	5	sync_mode	Sync Kommando aktiv
	6		reserviert
	7	slave_deactivated	1 wenn Slave vom Master deaktiviert
3	0 ... 6		reserviert
	7	ext_diag_overflow	Master oder Slave hat zu viele Diagnosedaten

Octet	Beschreibung
4	Masteradresse
5, 6	Ident-Nummer

Erweiterte Diagnose

Octet	Beschreibung
7	Länge der erweiterten Diagnose
8	SPS-Status
	0 - PLC state: Run 1 - PLC state: On 2 - PLC state: Run 4 - PLC state: Stop 5 - PLC state: Halt 6 - PLC state: Load
9	globale Fehlermeldungen
10	spezifische RIO Diagnose
11	reserviert
12	reserviert

Siehe dazu Betriebsanleitung "Inbetriebnahmehinweise für Feldbussysteme".

Alle Betriebsanleitungen können kostenlos vom Internet *http://www.schleicher-electronic.com* geladen werden.

6.9 Reaktionszeiten PROFIBUS-DP

Die Reaktionszeit wird definiert als die Gesamtzeit eines Nachrichtenzyklus zwischen Master und einem einzelnen Slave.
Ein Nachrichtenzyklus setzt sich zusammen aus einem Aufforderungstelegramm an den Slave, einzuhaltenden Busruhezeiten und der Antwortzeit des Slaves.
Die Buszykluszeit ergibt sich aus der Addition der Nachrichtenzyklen.

Um die Reaktionszeit zu berechnen, kann folgende Berechnungsvorschrift verwendet werden:

12 MBaud	28µs + 1µs/zu übertragendes Datenbyte
1.5 MBaud	224µs + 7µs/zu übertragendes Datenbyte

Beispiel:

10 Busknoten mit jeweils 8 Byte Ausgangsdaten und 8 Byte Eingangsdaten

12 MBaud:

28 + 8 + 8 = 44µs	Reaktionszeit
44 * 10 = 440µs	Buszykluszeit

1.5MBaud:

224 + (7*8) + (7*8) = 336µs	Reaktionszeit
336 * 10 = 3.4ms	Buszykluszeit

Addiert werden muss eine herstellerspezifische Laufzeit im DP-Master, typisch 1 - 3ms.

Also dauert ein Buszyklus, in dem alle Slaves einmal angsprochen werden, bei 12 MBaud ca. 2 - 4 ms.

7 Fehlermeldungen

Bei Auftreten eines SPS-Fehlers wird die SPS angehalten. Es werden alle Ausgänge abgeschaltet, d.h. auf Null gesetzt. Die Eingänge werden nicht weiter an die SPS gesendet. Die Diagnose in der Steuereinheit wird weiterhin betrieben, die Diagnosemeldungen werden übertragen.

Die SPS-Fehler werden bei SPS-Start oder SPS-Reset gelöscht.

Detaillierte Informationen zu Fehlermeldungen und Warnungen sind auch über MULTIPROG erhältlich.

Fehler-Nr. (HEX)	**Fehlergruppe 1 – Task-Kommunikation**
0021	Interner Fehler: Fehler in der Systemtask für die Speicherverwaltung
0031	Interner Fehler: Fehler in der Systemtask für die Kommunikation
0041	Interner Fehler: Fehler in der Systemtask für das Debugging
0051	Interner Fehler: Fehler in der Root-Task
0061	Interner Fehler: Fernsteuerung abgeschaltet

Fehler-Nr. (HEX)	**Fehlergruppe 2 – Code Generierung**
0012	Interner Fehler: Kein Programmcode für Modul
0022	Speicherfehler: Zuwenig Speicher für SPS !
0032	Anweisung nicht implementiert oder ungültiger Datentyp! -> Anweisung nicht verwenden oder Datentyp zusammen mit dieser Anweisung verwenden.
0042	Programmcode mehr als 64 KByte !
0052	Label fehlt !
0062	Interner Fehler: Fehler bei Parameterübergabe !
0072	Fehler in Verschachtelungstiefe ! Zuviele schliessende Klammern !
0082	Fehler bei Verschachtelung !
0092	Interner Fehler: Übergabeparameter fehlt !
00A2	Datentypkonvertierung fehlt !
00B2	Datentypkonvertierung für Akkumulator fehlt !
00C2	Maximale Anzahl Fehler erreicht ! Code-Generierung abgebrochen !
00D2	Fehler bei der Code-Generierung in POE !
00E2	Interner Fehler: Fehler bei der Parameterübergabe !
00F2	Interner Fehler: Operand nicht implementiert oder Bereichsüberschreitung !
0102	Unzulässiger Datentyp für Akkumulator (ANYNUM erwartet) !
0112	Unzulässiger Datentyp für Akkumulator (ANYBIT erwartet) !
0122	Interner Fehler: Ungültiger Datentyp für Akkumulator !
0132	Interner Fehler: Angewähltes Indexregister ungültig !
0142	Interner Fehler: Unzulässiger Akkumulator-Datentyp !
0152	Interner Fehler: Nicht alle Akkumulator-Werte, die zwischengespeichert wurden (pushed), wurden zurück gelesen (popped) !
0162	Interner Fehler: Maximal Anzahl von Akkumulator-Werten, die zwischengespeichert werden können (pushed), überschritten !
0172	Interner Fehler: Akkumulator-Wert wurde zurück gelesen (popped), der nicht zwischengespeichert wurde (pushed) !
0182	Ungültige Version für Ablaufsprache !
0192	Keine AS-Daten für Projekt ! SPS Reset wurde ausgeführt !

Fehler-Nr. (HEX)	Fehlergruppe 3 - Speicherverwaltung
0013	Interner Fehler: Ungültige POE !
0023	Interner Fehler: Unbekannter POE-Typ !
0033	Speicherfehler: Zuwenig SPS-Speicher zum Einfügen der POE !
0043	Speicherfehler: POE größer als 64 KByte beim Einfügen !
0053	Warnung: POE belegt mehr als 80 % des Speichers für POEs !
0063	Warnung: POE kann nicht eingefügt werden, weil kein Projekt vorhanden ist !
0073	Warnung: SPS-Projekt belegt mehr als 80 % des Programmspeichers !
0083	Es wurde versucht ein POE einzufügen, die nicht zu dem Projekt auf der SPS gehört !
0093	Interner Fehler: Fehler in der Speicherverwaltung !
00A3	Warnung: Aktuelle Projektgröße n KByte !
00B3	Interner Fehler: Fehler beim Einfügen der POE !
00E3	Interner Fehler: Ungültiger POE-Typ !
00F3	Interner Fehler: Speicherreorganisation nicht möglich !
0103	Warnung: Aktuelle POE-Größe n Bytes !
0113	Warnung: Interner Fehler: PG mehrfach definiert !
0123	Warnung: Interner Fehler: SPG mehrfach definiert !
0133	Speicherfehler bei POE-Initialisierungsdaten !
0143	Speichern der Checksumme für RETAIN-Daten fehlgeschlagen !
0153	Warnung: Interner Fehler: FB mehrfach definiert !
01B3	Interner Fehler: Nicht alle POEs gesendet !
01C3	Interner Fehler: Programmspeicher nicht definiert !
01D3	Interner Fehler: Ungültige FB-Nummer !
01E3	Interner Fehler: Ungültige PG-Nummer !
01F3	Interner Fehler: Ungültige SPG-Nummer !

Fehler-Nr. (HEX)	Fehlergruppe 4 - SPS-Anwenderfehler
0014	Interner Fehler: Ungültige Funktion oder ungültiger Funktionsbaustein !
0024	Ungültige Firmwarefunktion oder ungültiger Firmwarefunktionsbaustein !
0034	Interner Fehler: Ungültiges Programm !
0044	Warnung: Interner Fehler: Wechsel in ungültigen Modus !
0054	Interner Fehler: Unbekannter Systemmodus !
0074	Division durch 0 !
00A4	Warnung: Funktion PLC_STOP wurde ausgeführt, SPS gestoppt !
00B4	Bus-Zugriffsfehler ! Ungültige Speicheradresse !
00C4	Stack-Überlauf !
00D4	Interner Fehler: Nicht implementierter CPU-Befehl !
00E4	Interner Fehler: Interrupt nicht initialisiert !
00F4	Interner Fehler: Falscher Interrupt !
0104	Systemfehler in einem Modul !
0114	Systemfehler in Modul-Zeile !
0124	Fehler bei indirektem Variablenzugriff !
0134	Task-Watchdog ! Laufzeitüberschreitung !
0144	Interner Fehler: Fehler in Task-Konfiguration !
0154	Fehler in I/O-Konfiguration !
0164	Warnung: Warmstart nicht möglich ! Kaltstart ausgeführt !
0174	Warnung: Kein RETAIN-Speicherbereich !
0184	CPU überlastet ! SPS-Projekt benötigt zuviel CPU-Rechenzeit !
0194	Initialisierung des I/O-Treibers fehlgeschlagen !
01A4	Interner Fehler: Unerwarteter Breakpoint !
01B4	Unbekannter I/O-Treiber !
01C4	Interner Fehler: Watchdog in Systemtask !
01D4	Interner Fehler: Fehler in Datenkonfiguration !
01E4	Interner Fehler: Fehler in RETAIN-Datenkonfiguration !
0204	Interner Fehler: Kritischer Fehler in Floating-Point-Einheit !
0214	Interner Fehler: Fataler Fehler !

0224	String-Fehler: Falscher Zugriff auf Zeichenkette !
0234	String-Fehler !
0244	String-Fehler: Ausgabezeichenkette ist zu kurz !
0254	String-Fehler: Eingabezeichenkette ist zu kurz !
0264	String-Fehler: Ungültiger Eingabeparameter !
0274	String-Fehler: Zweite Zeichenkette ist identisch mit Ausgabezeichenkette !
0284	String-Fehler: Ungültiger Vergleich !
0294	String-Fehler: Nicht unterstützter Datentyp für Konvertierung !
02A4	String-Fehler: Fehler im Format-String !
02B4	String-Fehler: Ungültiger Wert für Format-String !
02C4	String-Fehler: Fehler bei Konvertierung !

Fehler-Nr. (HEX)	**Fehlergruppe 5 – Debug**
0025	Operand nicht implementiert oder Bereichsüberschreitung !
0035	Falsche oder fehlende Trigger-Bedingung !
0045	Kein Speicher verfügbar !
0055	Nicht unterstützter Datentyp !
0065	Interner Fehler: Breakpoint in Funktion oder Funktionsbaustein nicht möglich ! (nur bei Single-Step möglich)
0075	Warnung: Breakpoint in dieser Zeile nicht möglich !
0095	Warnung: Schreibzugriff für Variable nicht erlaubt !
00B5	Zuviele Powerflow-Adressen !
00C5	Powerflow nicht möglich !
00D5	Warnung: Interner Fehler: Fehler in Powerflow-Liste (ungültige Einträge) !
00F5	Interner Fehler: Ungültiger Listentyp !
0115	Keine Debug-Informationen für POE !
0125	Keine Debug-Information für Wortnummer !
0135	Debug-Funktion nicht verfügbar wenn Programm im (E)EPROM !
0145	Interner Fehler: Code für POE fehlt !
01A5	Interner Fehler: Breakpoints in PG nicht möglich !
01B5	Interner Fehler: Breakpoints in SPG nicht möglich !
01D5	Interner Fehler: Force-Liste nicht verfügbar !
0205	Interner Fehler: Debug-Task nicht installiert !

Fehler-Nr. (HEX)	**Fehlergruppe 6 - Systemdatenaustausch**
0016	Warnung: Interner Fehler: Ungültiger Modultyp !
0026	Zugriff auf Initialisierungsdatei fehlgeschlagen !
0036	Interner Fehler: Zuviele Daten angefordert !
0046	Interner Fehler: Zuwenig Daten gesendet !
0056	Interner Fehler: Falsche Online-Dienstanforderung !
0066	Installation des I/O-Treibers fehlgeschlagen !
0076	Instanzierung des I/O-Treibers fehlgeschlagen !
0086	Fehler beim Dateizugriff !
0096	Instanzierung fehlgeschlagen !
00A6	Interner Fehler: Datengruppe größer als 64 KByte !
00B6	Speicherfehler: Zuwenig dynamischer Speicher !
00C6	Interner Fehler: Fehlerhafter Refresh der RETAIN-Daten !
00D6	Gerät zur Speicherung der RETAIN-Daten nicht verfügbar !
00E6	Interner Fehler: Ungültige Gruppe für RETAIN-Daten !
00F6	Interner Fehler: (PDD) Doppelte Datentypdefinition !
0106	Interner Fehler: (PDD) Datentypdefinition (Subtyp) nicht gefunden !
0116	Interner Fehler: (PDD) Unbekannter Datentyp !
0126	Interner Fehler: (PDD) Symbol nicht gefunden !
0136	Interner Fehler: (PDD) Symbol existiert mehrfach !
0146	Interner Fehler: (PDD) Operandenbereich überschritten !
0156	Interner Fehler: (PDD) Kein Speicher für schnellen symbolischen Zugriff (nur Information) !
0166	Unbekannter I/O-Treiber !

0176	ZIP-Gerät nicht installiert !
0186	Datei-Gerät nicht installiert !

Fehler-Nr. (HEX)	**Fehlergruppe 7 – I/O-Treiber-Fehler**
0017	Board nicht instanziiert !
0027	Board nicht zulässig !
0037	Eingangsgruppe nicht passend !
0047	Ausgangsgruppe nicht passend !
0057	Board nicht gefunden !
0067	Fehler beim Lesen der Eingänge !
0077	Fehler beim Lesen der Ausgänge !
0087	Interner Fehler: Semaphoren können nicht erzeugt werden !
0097	Ungültige Speichergröße !
00A7	Ungültige Board-Adresse !

Fehler-Nr. (HEX)	**Fehlergruppe 15 – Firmware-Fehler**
00BF	PLC Firmware-Fehler
00CF	CAN Firmware-Fehler

8 Technische Daten und Abmessungen

.

Technische Daten Netzteil XNT24	
Spannungsversorgung	
Betriebsspannung	DC 24 V ± 20%
Restwelligkeit	max. 5%
Leistungsaufnahme	< 3 W (mit XCS20C oder XCS20P, ohne I/O-Module)
Gewicht	160 g (mit Verpackung)
Digitale Eingänge	
Anzahl	2
Schaltpegel	H-Pegel = +15 V bis +30 V L-Pegel = -30 V bis +5 V
Eingangsstrom	min. H-Pegel (+15 V) ≥ 2 mA max. L-Pegel (+5 V) ≤ 2 mA typ. (+24 V) 8 mA max. (+30 V) ≤ 15 mA
Signalverzögerung	< 300 µs (Hardware)
Triggerung	Flanken-Triggerung

Technische Daten Steuereinheit	XCS20C	XCS20P
Prozessortakt	24 MHz	24 MHz
Anzahl der anreihbaren I/O-Module.	8	8
Max. Anzahl Ein-/Ausgangsdaten (Bytes)	64/64	64/64
Serielle Schnittstellen	2 x RS232 (eine auf Netzteil)	2 x RS232 (eine auf Netzteil)
Feldbusanschluss	CANopen	PROFIBUS
Gewicht	160 g (mit Verpackung)	160 g (mit Verpackung)
Speicher		
RAM	1 MB, davon 256 KB für SPS-Programme und 64 KB für RETAIN-Variablen	
Flash-Speicher	1024 KB, je 512 KB für Betriebssystem und Anwenderdaten (Anwenderdaten: Bootprojekt 192 KB, Programmquelle 256 KB, CANopen-Konfiguration 64 KB)	
Pufferelement	Vanadium-Pentoxid-Lithium-Zelle 3V / 50 mAh + SuperCAP	
Pufferzeit	min. 3 Monate (Setzt eine volle Aufladung der Zelle voraus, die mit einer ununterbrochenen Betriebszeit der Steuereinheit von 4 h erreicht wird.)	
Hardware		
CPU	Motorola MC68LC302	
Real-Time Clock	Batteriegepuffert mit Kalender und Schaltjahr, Auflösung: 1s	
Serielle Schnittstelle	RS232	
SPS-Eigenschaften		
Bearbeitungszeit für 1000 Anweisungen		
Datentyp BOOL	2,62 ms	
Datentyp BYTE	2,77 ms	
Datentyp WORD	1,94 ms	
Datentyp DWORD	2,02 ms	
Funktionsbausteine	max. 256 Firmwarefunktionen und Funktionsbausteine	
Betriebssystem	Multitask-Betriebssystem	
Anzahl der Tasks	16	
Taskzykluszeiten	≥ 2ms (geradzahlig)	

Speicherverwaltung	dynamisch
Zeiten und Zähler	beliebig viele, programmierbar 2 ms .. 290 h (Anzahl nur durch Speicherauslastung begrenzt)
Programmierung	Programmiersoftware MULTIPROG nach IEC 61131-3

Die zugrundeliegende Produktnorm ist EN 61131-2.

Klimatische Bedingungen		
Betriebsumgebungstemperatur T_B	0°C bis +55°C, senkrechter Einbau, freie Luftzirkulation	
Lagertemperatur	-25°C bis +70°C	
Relative Luftfeuchte	10% bis 95%, keine Betauung	
Luftdruck im Betrieb	860 hPa bis 1060 hPa	
Mechanische Festigkeit		
Schwingen	5 Hz ... 9 Hz konstante Amplitude 3,5 mm 9 Hz … 150 Hz konstante Beschleunigung 1 g	EN 60068-2-6
Elektrische Sicherheit		
Schutzart	IP 20	EN 60529
Luft-/Kriechstrecken		EN 61131-2 und EN 50178
Überspannungskategorie	II	
Verschmutzungsgrad	2	
Elektromagnetische Verträglichkeit		
Elektrostatische Entladung (ESD)	8 kV Luftentladung, 4 kV Kontaktentladung	EN 61000-4-2
Störfelder, HF-Felder	Feldstärke 10 V/m, 26 ... 1000, 1400 ... 2000 MHz	EN 61000-4-3
Schnelle Transienten (Burst)	2 kV auf DC-Versorgungsleitungen, 1 kV auf Signal- und seriellen Schnittstellenleitungen	EN 61000-4-4
Energiereiche Transienten (Surge)		EN 61000-4-5
DC-Versorgung	1 kV asymmetrisch (CM) 0,5 kV symmetrisch (DM)	
Analoge oder DC-Ein-/Ausgänge, ungeschirmt	0,5 kV asymmetrisch (CM) / symmetrisch (DM)	
Alle geschirmten Leitungen	1 kV asymmetrisch (CM)	
Störaussendung, HF-Felder	30 ... 230 MHz mit 40 dB (µV), 230 ... 1000 MHz mit 47 dB (µV)	EN 55011 Grenzwertklasse A, Gruppe1

Anschlusstechnik	
Klemmen	Schraubklemmen steckbar
Anschlussquerschnitte	
Eindrähtig oder feindrähtig	1 x 0,14 mm² bis 2,5 mm² oder 2 x 0,14 mm² bis 0,75 mm²
Abisolierlänge	max. 8 mm
Feindrähtig mit Aderendhülse nach DIN 46228	1 x 0,25 mm² bis 2,5 mm² oder 2 x 0,25 mm² bis 0,5 mm²
Maximales Anzugsdrehmoment der Schraubklemmen	0,5 bis 0,6 Nm

Abmessungen

Netzteil

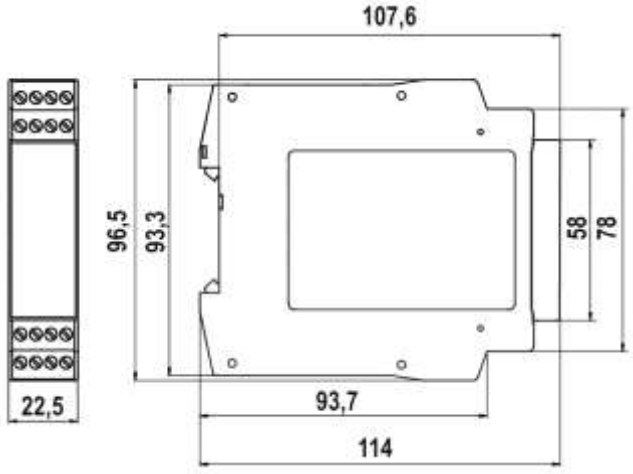


Abbildung 19: Abmessungen Netzteil

Steuereinheiten

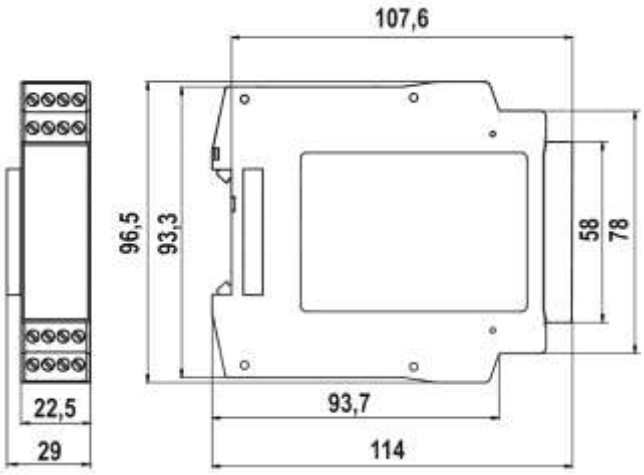


Abbildung 20: Abmessungen Steuereinheit

9 Anhang

9.1 Warenzeichenvermerke

- MS-DOS ist ein eingetragenes Warenzeichen der Microsoft Corporation.
- WINDOWS ist ein eingetragenes Warenzeichen der Microsoft Corporation.
- IBM ist ein eingetragenes Warenzeichen der International Business Machines.
- SIMATIC und SINEC sind eingetragene Marken der Siemens AG.
- DeviceNet ist ein eingetragenes Warenzeichen der Open DeviceNet Vendor Association (O.D.V.A.)
- CANopen ist ein eingetragenes Warenzeichen von CAN in Automation e.V.
- ProCANopen ist ein eingetragenes Warenzeichen von Vector Informatik GmbH
- CANalyzer ist ein eingetragenes Warenzeichen von Vector Informatik GmbH
- PROFIBUS ist ein eingetragenes Warenzeichen der PROFIBUS Nutzerorganisation

Alle anderen Warenzeichen oder Produktnamen sind eingetragene Warenzeichen der jeweiligen Firmen.

10 Sicherheitshinweise

Der im folgenden verwendete Begriff Automatisierungssysteme umfasst Steuerungen, sowie deren Komponenten (Module), andere Teile (wie z.B. Baugruppenträger, Verbindungskabel), Bediengeräte und Software, die für die Programmierung, Inbetriebnahme und Betrieb der Steuerungen genutzt wird. Die vorliegende Betriebsanleitung kann nur einen Teil des Automatisierungssystems (z.B. Module) beschreiben.

Die technische Auslegung der SCHLEICHER Automatisierungssysteme basiert auf der Produktnorm EN 61131-2 (IEC 61131-2) für speicherprogrammierbare Steuerungen. Für die Systeme und Geräte gilt grundsätzlich die CE-Kennzeichnung nach der EMV-Richtlinie 89/336/EWG und sofern zutreffend auch nach der Niederspannungsrichtlinie 73/23/EWG.

Die Maschinenrichtlinie 89/392/EWG ist nicht wirksam, da die in der Richtlinie genannten Schutzziele auch von der Niederspannungs- und EMV-Richtlinie abgedeckt werden.
Sind die SCHLEICHER Automatisierungssysteme Teil der elektrischen Ausrüstung einer Maschine, müssen sie vom Maschinenhersteller in das Verfahren zur Konformitätsbewertung einbezogen werden. Hierzu ist die Norm DIN EN 60204-1 zu beachten (Sicherheit von Maschinen, allgemeine Anforderungen an die elektrische Ausrüstung von Maschinen).

Von den Automatisierungssystemen gehen bei bestimmungsgemäßer Verwendung und ordnungsgemäßer Unterhaltung im Normalfall keine Gefahren in Bezug auf Sachschäden oder für die Gesundheit von Personen aus. Es können jedoch durch angeschlossene Stellelemente wie Motoren, Hydraulikaggregate usw. bei unsachgemäßer Projektierung, Installation, Wartung und Betrieb der gesamten Anlage oder Maschine, durch Nichbeachten von Anweisungen in dieser Betriebsanleitung und bei Eingriffen durch ungenügend qualifiziertes Personal Gefahren entstehen.

10.1 Bestimmungsgemäße Verwendung

Die Automatisierungssysteme sind nach dem Stand der Technik und den anerkannten sicherheitstechnischen Regeln gebaut. Dennoch können bei ihrer Verwendung Gefahren für LeIBund Leben des Benutzers oder Dritter bzw. Beeinträchtigungen von Maschinen, Anlagen oder anderen Sachwerten entstehen.

Das Automatisierungssystem darf nur in technisch einwandfreiem Zustand sowie bestimmungsgemäß, sicherheits- und gefahrenbewusst unter Beachtung der Betriebsanleitung benutzt werden. Der einwandfreie und sichere Betrieb der Steuerung setzt sachgemäßen Transport, sachgerechte Lagerung und Montage sowie sorgfältige Bedienung und Wartung voraus. Insbesondere Störungen, die die Sicherheit beeinträchtigen können, sind umgehend beseitigen zu lassen.

Die Automatisierungssysteme sind ausschließlich zur Steuerung von Maschinen und Anlagen vorgesehen. Eine andere oder darüber hinausgehende Benutzung gilt nicht als bestimmungsgemäß. Für daraus resultierende Schäden haftet der Hersteller nicht.

Zur bestimmungsgemäßen Verwendung der Automatisierungssysteme sind die in dieser Betriebsanleitung beschriebenen Anweisungen zum mechanischen und elektrischen Aufbau, zur Inbetriebnahme und zum Betrieb zu beachten.

10.2 Personalauswahl und -qualifikation

Alle Projektierungs-, Programmier-, Installations-, Inbetriebnahme-, Betriebs- und Wartungsarbeiten in Verbindung mit dem Automatisierungssystem dürfen nur von geschultem Personal ausgeführt werden (z.B. Elektrofachkräfte, Elektroingenieure).

Das Projektierungs- und Programmierpersonal muss mit den Sicherheitskonzepten der Automatisierungstechnik vertraut sein.

Das Bedienpersonal muss im Umgang mit der Steuerung unterwiesen sein und die Bedienungsanweisungen kennen.

Das Installations-, Inbetriebnahme- und Wartungspersonal muss eine Ausbildung besitzen, die zu Eingriffen am Automatisierungssystem berechtigt.

10.3 Projektierung, Programmierung, Installation, Inbetriebnahme und Betrieb

Das Automatisierungssystem ist in seiner Anwendung zumeist Bestandteil größerer Systeme oder Anlagen, in denen Maschinen gesteuert werden. Bei Projektierung, Installation und Inbetriebnahme der Automatisierungssysteme im Rahmen der Steuerung von Maschinen müssen deshalb durch den Maschinenhersteller und Anwender die Sicherheitsbestimmungen der Maschinenrichtlinie 89/392/EWG beachtet werden. Im spezifischen Einsatzfall geltende nationale Unfallverhütungsvorschriften wie z.B. VBG 4.0.

Alle sicherheitstechnischen Vorrichtungen der gesteuerten Maschine sind so auszuführen, dass sie unabhängig von der Steuerung funktionieren. Not-Aus-Einrichtungen müssen in allen Betriebsarten der Steuerung wirksam bleiben. Im Not-Aus-Fall müssen die Versorgungsspannungen aller von der Steuerung angesteuerten Schaltelemente abgeschaltet werden.

Es sind Vorkehrungen zu treffen, dass nach Spannungseinbrüchen und -ausfällen ein unterbrochenes Steuerungsprogramm ordnungsgemäß wieder aufgenommen werden kann. Dabei dürfen auch kurzzeitig keine gefährlichen Betriebszustände auftreten. Gegebenenfalls ist Not-Aus zu erzwingen.
Damit ein Leitungsbruch auf der Signalseite nicht zu undefinierten Zuständen in der Steuerung führen kann, sind bei der E/A-Kopplung hard- und softwareseitig entsprechende Sicherheitsvorkehrungen zu treffen. Einrichtungen der Steuerungstechnik und deren Bedienelemente sind so einzubauen, dass sie gegen unbeabsichtigte Betätigung ausreichend geschützt sind.

10.4 Wartung und Instandhaltung

Werden Meß- oder Prüfarbeiten am aktiven Gerät erforderlich, dann sind die Festlegungen und Durchführungsanweisungen der Unfallverhütungsvorschrift VBG 4.0 zu beachten. Es ist geeignetes Elektrowerkzeug zu verwenden.

Reparaturen an Steuerungskomponenten dürfen nur von SCHLEICHER autorisierten Reparaturstellen vorgenommen werden. Unbefugtes Öffnen und unsachgemäße Eingriffe oder Reparaturen können zu Körperverletzungen oder Sachschäden führen.

Vor Öffnen des Gerätes ist immer die Verbindung zum speisenden Netz zu trennen (Netzstecker ziehen oder Trennschalter öffnen).

Steuerungsmodule dürfen nur im spannungslosen Zustand gewechselt werden. Demontage und Montage sind gemäß der mechanischen Aufbaurichtlinien vorzunehmen.

Beim Auswechseln von Sicherungen dürfen nur Typen verwendet werden, die in den technischen Daten spezifiziert sind.

Beim Austausch von Batterien dürfen nur Typen verwendet werden, die in den technischen Daten spezifiziert sind. Batterien sind in jedem Fall nur als Sondermüll zu entsorgen.

10.5 Gefahren durch elektrische Energie

Nach Öffnen des Systemschrankes oder nach Entfernen des Gehäuses von Systemkomponenten werden bestimmte Teile des Automatisierungssystems zugänglich, die unter gefährlicher Spannung stehen können.

Der Anwender muss dafür sorgen, dass unbefugte und unsachgemäße Eingriffe unterbunden werden (z.B. verschlossener Schaltschrank).

Das Personal muss gründlich mit allen Gefahrenquellen und Maßnahmen zur Inbetriebnahme und Wartung gemäß den Angaben in der Betriebsanleitung vertraut sein.

10.6 Umgang mit verbrauchten Batterien

Die in den Automatisierungssystemen verwendeten Batterien sind, nach deren Verbrauchsende, dem Gemeinsamen Rücknahmesystem Batterien (GRS) oder öffentlich-rechtlichen Entsorgungsträgern zuzuführen.

Batterien sollen nur im entladenen Zustand zurückgegeben werden. Der entladene Zustand ist erreicht, wenn eine Funktionsbeeinträchtigung des Gerätes wegen unzureichender Batteriekapazität vorliegt.

Bei nicht vollständig entladenen Batterien muss Vorsorge gegen mögliche Kurzschlüsse getroffen werden. Das kann durch Isolieren der Batteriepole mit Klebestreifen erreicht werden.

11 Abbildungsverzeichnis

Abbildung 1: Frontansicht XNT24 9
Abbildung 2: Serielle Schnittstelle XNT24 10
Abbildung 3: Frontansicht XCS20C 11
Abbildung 4: Frontansicht XCS20P 11
Abbildung 5: LEDs XCS20C 12
Abbildung 6: Anschlüsse XCS20P 13
Abbildung 7: LEDs XCS20P 14
Abbildung 8: RESET-Taster XCS20C / XCS20P 15
Abbildung 9: Drehschalter XCS20C / XCS20P 16
Abbildung 10: Montagelage 18
Abbildung 11: Montageabstände 18
Abbildung 12: Verbindungskabel Steuereinheit-PC 21
Abbildung 13: Multi-Task-System 23
Abbildung 14: Watchdog 30
Abbildung 15: Betriebszustände 33
Abbildung 16: Bustopologie PROFIBUS-DP 63
Abbildung 17: Busverkabelung PROFIBUS-DP 66
Abbildung 18: Busverkabelung (Schirmung) PROFIBUS-DP 67
Abbildung 19: Abmessungen Netzteil 78
Abbildung 20: Abmessungen Steuereinheit 78

12 Index

A

Abisolierlänge 77
Abmessungen 78
Abschlusswiderstände
CANopen 61
Anschlüsse
Netzteil XNT24 9
Steuereinheit XCS20C 12
Steuereinheit XCS20P 13
Anschlussquerschnitte 77
Anschlusstechnik 77
Anzeigen
Steuereinheit XCS20C 12
Steuereinheit XCS20P 14

B

Batterie für Speicherpufferung 22
Baudrate 17
BETRIEB Betriebszustand 32
Betriebsartenschalter
Steuereinheit XCS20C 15
Steuereinheit XCS20P 15
Betriebszustand der SPS 32
Buskabel
PROFIBUS-DP 65
Bussegmentlänge
PROFIBUS-DP 65

C

CANopen
Abschlusswiderstände 61
Datenübertragungsrate 62
EDS-Dateien 61
Kabellänge 61
Knotennummer einstellen 61
Verkabelung 61

D

Datenbit 17
Datenübertragungsrate CANopen 62
Datenübertragungsrate und Leitungslänge
PROFIBUS-DP 65
DEFAULT-Task 24
Deklarieren
Task 27
Diagnose
am PROFIBUS-DP 69
Diagnosefunktionen 44
digitale Eingänge
auf dem Netzteil XNT24 10
Drehschalter
Steuereinheit XCS20C 15
Steuereinheit XCS20P 15

E

Echtzeituhr Setzen/Auslesen 42
EDS-Dateien für
CANopen 61
EIN Betriebszustand 32
Eingänge
auf dem Netzteil XNT24 10
Einstellen
Datenübertragungsrate CANopen 62
Knotennummer CANopen 61
Elektrische Sicherheit 77
Ereignis-Task 24

F

Fehlermeldungen 72

Firmware-Task Prioritäten **27**

H

HALT Betriebszustand **32**

I

I/O-Konfiguration **38**
I/O-Treiber
CANIO 37, 40
XDPIO 37, 40
IEC 61131-3
Konformität 58
Installationsrichtlinien **20**
Interrupt-Betrieb **43**

K

Kabellänge
CANopen 61
PROFIBUS-DP 65
Klimatische Bedingungen **77**
Knotennummer einstellen
CANopen 61
Kombikanäle **20**

L

Leitungslänge
CANopen 61
PROFIBUS-DP 65

M

Mechanische Festigkeit **77**
Mechanische Installation **18**
Merker Systemmerker **36**
Montageabstände **18**
Montagelage **18**

N

Netzteil XNT24
Abmessungen 78
allgemeines 9
Anschlüsse 9
digitale Eingänge 10
Technische Daten 76

P

Parametrierfunktionen **44**
Parität **17**
POE
Zuweisen zu Task 31
PROFIBUS-DP
Buskabel 65
Bussegmentlänge 65
Datenübertragunsrate und Leitungslänge 65
Diagnose am 69
GSD-Datei 68
Kabellänge 65
Leitungslänge 65
Reaktionszeiten 71
Verkabelung 66
Programm
Zuweisen zu Task 31
Programmiergerät
Anschluss eines PC 21
Programmierkabel **21**
Programmierung
Anschluss eines PC 21
Programmierung der SPS **21**
Prozessabbildspeicher **22**

R

Reaktionszeiten
PROFIBUS-DP 71
Reset-Taster
Steuereinheit XCS20C 15
Steuereinheit XCS20P 15

S

serielle Schnittstelle
auf dem Netzteil 10
auf der Steuereinheit 12, 13
serielle Schnittstellen **17**
Sicherheitshinweise **80**
Bestimmungsgemäße Verwendung 80
Darstellung Warnhinweise 5
Inbetriebnahme 81
Installation 81
Instandhaltung 81
Not-Aus-Einrichtung 81
Personalauswahl 80
Programmierung 81
Projektierung 81
Unfallverhütungsvorschrift 81
Wartung 81
Speicher
Prozessabbildspeicher 22
Pufferung 22
SPS-Programm 22
SPG siehe Systemprogramme **25**
SPS
Adressen 35
Betriebsarten 32
Programmierung 21
Startverhalten 32
Steuereinheit XCS20C
Abmessungen 78
allgemeines 11
Anschlüsse 12
Betriebsartenschalter 15
Drehschalter 15
LED-Anzeigen 12
Reset-Taster 15
Technische Daten 76
Steuereinheit XCS20P
Abmessungen 78
allgemeines 11
Anschlüsse 13
Betriebsartenschalter 15
Drehschalter 15
LED-Anzeigen 14
Reset-Taster 15
Technische Daten 76
STOP Betriebszustand **32**
Stopbit **17**
Systemmerker **36**
Systemprogramme SPG's **25**

System-Task **25**

T

Task
Anwender-Task 23
DEFAULT-Task 24
Deklarieren 27
Ereignis-Task 24
Firmware-Task Prioritäten 27
Information 28
Prioritäten 27
Prioritätsstufe 23
System-Task 25
Task-Information 28
Überwachungstask 23
Watchdog 30
zyklische Task 24
Technische Daten **76**
Tragschiene **19**
Tragschienenmontage **19**

V

Verbindungskabel Steuereinheit - PC **21**
Verkabelung
CANopen 61
PROFIBUS-DP 66

W

Warenzeichenvermerke **79**
Watchdog **30**

Z

Zuweisen
POE zu Task 31
Programm zu Task 31
zyklische Task **24**